Контрольно-оценочные средства для проведения текущего контроля
по МДК.01.02 Прикладное программирование
(3 курс, 5 семестр 2022-2023 уч. г.)

Текущий контроль №1

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита
Задание №1

Разобрать текст программы, написанной на языке С++

```
#include <iostream.h>
#include <math.h>
#include <stdio.h>
#define PI 3.14
void main()
{float r,v,s;
cout<<"Vvedite r\n";
cin>>r;
v=(3*PI*r*r*r)/4;
s=4*PI*r*r;
cout<<"s="<<s<<endl;
cout<<"v="<<v<<endl;
}
```

Оценка	Показатели оценки
3	Ввести правильно в среду С++ Builder
4	Ввести и отладить программу, показать результат работы
5	Отладить программу, объяснить текст и результат

Задание №2

Разобрать текст программы, написанной на языке С++

```
#include <iostream.h>
#include <math.h>
#include <stdio.h>
#define PI 3.14
void main()
{float r,v,s;
cout<<"Vvedite r\n";
cin>>r;
v=(3*PI*r*r*r)/4;
s=4*PI*r*r;
```

```
cout<<"s="<<s<<endl;
cout<<"v="<<v<<endl;
}
```

Оценка	Показатели оценки
3	Ввести правильно в среду C++ Builder
4	Ввести и отладить программу, показать результат работы
5	Отладить программу, объяснить текст и результат

Текущий контроль №2

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита
Задание №1

Написать программы на языке C++
1. Написать программу вычисления площади треугольника, если известны длины двух его сторон

и
величина угла между этими сторонами.
S= a*b*sin(u*p/180)/2

Оценка	Показатели оценки
3	Определение типов переменных Задание типов переменных
4	Определение типов переменных Задание типов переменных Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Определение типов переменных Задание типов переменных Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Задание №2

Написать программы на языке С++
1. Написать программу вычисления площади треугольника, если известны длины двух его сторон

и
величина угла между этими сторонами.
S= a*b*sin(u*p/180)/2

Оценка	Показатели оценки
3	Определение типов переменных Задание типов переменных
4	Определение типов переменных Задание типов переменных Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Определение типов переменных Задание типов переменных Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Задание №3

Написать программы на языке С++
1. Написать программу вычисления площади треугольника, если известны длины двух его сторон

и
величина угла между этими сторонами.
S= a*b*sin(u*p/180)/2

Оценка	Показатели оценки
3	Определение типов переменных Задание типов переменных
4	Определение типов переменных Задание типов переменных Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма

5	Определение типов переменных Задание типов переменных Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Текущий контроль №3

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита
Задание №1

Написать программы на языке С++
Написать программу решения квадратного уравнения. Программа должна проверять

правильность
исходных данных и в случае, если коэффициент при второй степени неизвестного равен нулю,

выводить
соответствующее сообщение.

Оценка	Показатели оценки
3	Определение типов переменных Задание типов переменных Определить компоненты и их свойства
4	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Задание №2

.Напишите программу , которая запрашивает у пользователя номер дня недели, затем выводит

название

дня недели или сообщение об ошибке, если введены неверные данные

Оценка	Показатели оценки
3	Определение типов переменных Задание типов переменных Определить компоненты и их свойства
4	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Задание №3

Напишите программу , которая запрашивает у пользователя номер дня недели, затем выводит

название

дня недели или сообщение об ошибке, если введены неверные данные

Оценка	Показатели оценки
3	Определение типов переменных Задание типов переменных Определить компоненты и их свойства

4	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Текущий контроль №4

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита
Задание №1

1.Написать программу, которая выводит таблицу квадратов десяти первых положительных чисел.

Оценка	Показатели оценки
3	Определение типов переменных Задание типов переменных Определить компоненты и их свойства
4	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Задание №2

1.Написать программу, которая выводит таблицу квадратов десяти первых положительных чисел.

Оценка	Показатели оценки
3	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства
4	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Задание №3

1. Написать программу, которая выводит таблицу значений функции y=-2,4x 2 +5x-3 в диапазоне

от –2 до
+2, с шагом 0,5

Оценка	Показатели оценки
3	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства
4	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма

5	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Текущий контроль №5

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита
Задание №1

Написать программу, которая определяет максимальное число из введенной с клавиатуры последовательности положительных чисел.(длина последовательности неограниченна)

Оценка	Показатели оценки
3	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства
4	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Задание №2

Написать программу, которая определяет максимальное число из введенной с клавиатуры последовательности положительных чисел.(длина последовательности неограниченна)

Оценка	Показатели оценки

3	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства
4	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Задание №3

Написать программу, которая определяет максимальное число из введенной с клавиатуры последовательности положительных чиссл.(длина последовательности неограниченна)

Оценка	Показатели оценки
3	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства
4	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Текущий контроль №6

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита
Задание №1

Создать двумерный массив из случайно сгенерированных чисел (от 1 до 10)
Подсчитать сумму нечетных чисел элементов массива. Выдать на экран массив и найденную сумму.

Оценка	Показатели оценки
3	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства
4	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Задание №2

Создать двумерный массив из случайно сгенерированных чисел (от 1 до 10)
Подсчитать сумму нечетных чисел элементов массива. Выдать на экран массив и найденную сумму.

Оценка	Показатели оценки
3	Разработка системы тестов, охватывающих: средние значения; граничные условия входных данных; граничные условия выходных данных.

4	Разработка системы тестов, охватывающих: средние значения; граничные условия входных данных; граничные условия выходных данных. Проверка программы на разработанной системе тестов
5	Разработка системы тестов, охватывающих: средние значения; граничные условия входных данных; граничные условия выходных данных. Проверка программы на разработанной системе тестов Заключение о работоспособности алгоритма, реализованном в программе

Текущий контроль №7

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита

Текущий контроль №8

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита
Задание №1

Написать функцию Dohod, которая вычисляет доход по вкладу. Исходными данными для

функции
являются: величина вклада, процентная ставка (годовых) и срок вклада (количество дней)

Оценка	Показатели оценки
3	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства

4	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Текущий контроль №9

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита

Текущий контроль №10

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита

Текущий контроль №11

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита
Задание №1

В качестве комплексного примера рассмотрим класс Point, который позволяет сформулировать

точку на
экране компьютера. Поместим описание класса в файл с именем point.h:

```
// Файл point.h
#ifndef POINT_H
#define POINT_H 1
class Point //класс для определения точки на экране
{
protected: //защищенный статус доступа к элементам данных
int x; //координата x точки
int y; //координата y точки
```

```
// Прототипы методов:
public: //общедоступный статус доступа
Point (int, int); //конструктор
int putx() t //доступ к x
int puty(); //доступ к y
void show0; //изобразить точку на экране
void move (int,int); //переместить точку
private: //собственный статус доступа
void hide(); //убрать изображение точки
};
#endif
```

Поскольку описание класса Point планируется использовать при описании других классов, то для предотвращения недопустимого дублирования описания класса в текст включены три директивы препроцессора #definePOINTH, #definePOINT_H 1 и #endif.

Компонентами класса Point являются два элемента данных x и y с защищенным статусом доступа, пять
общедоступных методов и один метод с собственным статусом доступа. Методы в описании класса
представлены своими прототипами

Оценка	Показатели оценки
3	Ввести текст программы в среду C++ Разработка системы тестов, охватывающих: средние значения; граничные условия входных данных; граничные условия выходных данных.
4	Разработка системы тестов, охватывающих: средние значения; граничные условия входных данных; граничные условия выходных данных. Проверка программы на разработанной системе тестов

5	Разработка системы тестов, охватывающих: средние значения; граничные условия входных данных; граничные условия выходных данных. Проверка программы на разработанной системе тестов Заключение о работоспособности алгоритма, реализованном в программе

Текущий контроль №12

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита
Задание №1

Выполним внешнее описание методов класса, разместив описания в файле
Point.cpp

```
//Файл point.cpp - описание методов
#ifndef POINT_CPP
#define POINT_CPP 1
#include <graphics.h>// прототипы функций графической библиотеки
#include "f:\POS_C\PRIMER\point.h" // описание класса Point
// Конструктор:
Point :: Point {int xl=0, int yl=())
{
x=xl;
y=yl;
}
//Метод доступа к x:
int Point :: putx()
{
return x;

}
//Метод доступ к y:
int Point :: puty()
{
return y;
}
//Метод изображения точки на экране:
void Point :: show(void)
{
```

```
putpixel(x,y,getcolor()) ;
}
// Метод удаления точки с экрана:
void Point :: hide(void)
{
putpixel(x,y,getbkcolor());
}
// Метод перемещения точки в новое место экрана:
void Point :: move(int xn=0, int yn=0)
{
hide();
x=xn;
y=yn;
show();
}
#endif
```

Для получения изображения точки на экране в программу должен быть включен заголовочный файл graphics.h, В данном файле находятся прототипы графических функций.

Оценка	Показатели оценки
3	Ввести текст программы в среду C++ Определение типов переменных Задание типов переменных. Определить компоненты и их свойства
4	Ввести текст и отладить программу, получить результат Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Отладить программу и объяснить ее работу Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Задание №2

Выполним внешнее описание методов класса, разместив описания в файле
Point.cpp

```
//Файл point.cpp - описание методов
#ifndef POINT_CPP
#define POINT_CPP 1
#include <graphics.h>// прототипы функций графической библиотеки
#include "f:\POS_C\PRIMER\point.h" // описание класса Point
// Конструктор:
Point :: Point {int xl=0, int yl=())
{
x=xl;
y=yl;
}
//Метод доступа к x:
int Point :: putx()
{
return x;

}
//Метод доступ к y:
int Point :: puty()
{
return y;
}
//Метод изображения точки на экране:
void Point :: show(void)
{
putpixel(x,y,getcolor()) ;
}
// Метод удаления точки с экрана:
void Point :: hide(void)
{
putpixel(x,y,getbkcolor());
}
// Метод перемещения точки в новое место экрана:
void Point :: move(int xn=0, int yn=0)
{
hide();

x=xn;
y=yn;
show();
}
#endif
```

Для получения изображения точки на экране в программу должен быть включен заголовочный файл graphics.h, В данном файле находятся прототипы графических функций.

Оценка	Показатели оценки

3	Ввести текст программы в среду С++ Определение типов переменных Задание типов переменных. Определить компоненты и их свойства
4	Ввести текст и отладить программу, получить результат Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Отладить программу и объяснить ее работу Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Задание №3

Выполним внешнее описание методов класса, разместив описания в файле
Point.cpp

```
//Файл point.cpp - описание методов
#ifndef POINT_CPP
#define POINT_CPP 1
#include <graphics.h>// прототипы функций графической библиотеки
#include "f:\POS_C\PRIMER\point.h" // описание класса Point
// Конструктор:
Point :: Point {int xl=0, int yl=())
{
x=xl;
y=yl;
}
//Метод доступа к x:
int Point :: putx()
{
return x;

}
```

```
//Метод доступ к y:
int Point :: puty()
{
return y;
}
//Метод изображения точки на экране:
void Point :: show(void)
{
putpixel(x,y,getcolor()) ;
}
// Метод удаления точки с экрана:
void Point :: hide(void)
{
putpixel(x,y,getbkcolor());
}
// Метод перемещения точки в новое место экрана:
void Point :: move(int xn=0, int yn=0)
{
hide();



x=xn;
y=yn;
show();
}
#endif
```

Для получения изображения точки на экране в программу должен быть включен заголовочный файл graphics.h, В данном файле находятся прототипы графических функций.

Оценка	Показатели оценки
3	Ввести текст программы в среду C++ Определение типов переменных Задание типов переменных. Определить компоненты и их свойства
4	Ввести текст и отладить программу, получить результат Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма

5	Отладить программу и объяснить ее работу Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Текущий контроль №13

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита
Задание №1

Программа, содержащая в своем составе главную функцию, будет выглядеть следующим образом:

// Точка на экране

```
#include <iostream.h>

#include <graphics.h> //прототипы графических функций

#include <conio.h> //прототип функции getch()

#include "f:\POS_C\PRIMER\point.cpp" //описание класса Point

int main()

{

Point t(100,150); //создана невидимая точка t(x=100,y=150)

Point tl(200,200); //создана невидимая точка tl(X=200,Y=200)

// Инициализация графики

int a=DETECT,b;

initgraph(&a,&b,"F:\\BC5\\bgi");

t.show(); . / / показать точку t
```

```
cout « "\n коор-ты точки t: x=" « t.putx() « ",y=" « t.puty();
getch(); //ждать нажатия клавиши

tl.show(); //показать тоочку tl

cout « "\n координаты точки tl: x=" « tl.putx() « ",y=" «

tl.puty();

getch();

t.move(150,300); //переместить точку t(x=150,y=300)

cout « "\n новые координаты t: x=" « t.putx() « ",y=" «

t.puty(); getch();

closegraph(); //закрыть графический режим

}
```

В программе используется функция *getch(),* которая позволяет остановить выполнение программы до нажатия на клавиатуре любой клавиши. Прототип этой функции находится в файле conio.h.

Оценка	Показатели оценки
3	Ввести текст программы в среду в С++ Определение типов переменных Задание типов переменных. Определить компоненты и их свойства
4	Ввести текст и отладить программу получить результат Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма

5	Отладить программу и объяснить ее работу Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Задание №2

Программа, содержащая в своем составе главную функцию, будет выглядеть следующим образом:

// Точка на экране

```
#include <iostream.h>

#include <graphics.h> //прототипы графических функций

#include <conio.h> //прототип функции getch()

#include "f:\POS_C\PRIMER\point.cpp" //описание класса Point

int main()

{

Point t(100,150); //создана невидимая точка t(x=100,y=150)

Point tl(200,200); //создана невидимая точка tl(X=200,Y=200)

// Инициализация графики

int a=DETECT,b;

initgraph(&a,&b,"F:\\BC5\\bgi");

t.show(); . / / показать точку t

cout « "\n коор-ты точки t: x=" « t.putx() « ",y=" « t.puty();
getch(); //ждать нажатия клавиши

tl.show(); //показать тоочку tl
```

```
cout « "\n координаты точки tl: x=" « tl.putx() « ",y=" «

tl.puty();

getch();

t.move(150,300); //переместить точку t(x=150,y=300)

cout « "\n новые координаты t: x=" « t.putx() « ",y=" «

t.puty(); getch();

closegraph(); //закрыть графический режим

}
```

В программе используется функция *getch(),* которая позволяет остановить выполнение программы до нажатия на клавиатуре любой клавиши. Прототип этой функции находится в файле conio.h.

Оценка	Показатели оценки
3	Разработка системы тестов, охватывающих: средние значения; граничные условия входных данных; граничные условия выходных данных.
4	Разработка системы тестов, охватывающих: средние значения; граничные условия входных данных; граничные условия выходных данных. Проверка программы на разработанной системе тестов

5	Разработка системы тестов, охватывающих: средние значения; граничные условия входных данных; граничные условия выходных данных. Проверка программы на разработанной системе тестов Заключение о работоспособности алгоритма, реализованном в программе

Задание №3

Программа, содержащая в своем составе главную функцию, будет выглядеть следующим образом:

// Точка на экране

```
#include <iostream.h>

#include <graphics.h> //прототипы графических функций

#include <conio.h> //прототип функции getch()

#include "f:\POS_C\PRIMER\point.cpp" //описание класса Point

int main()

{

Point t(100,150); //создана невидимая точка t(x=100,y=150)

Point tl(200,200); //создана невидимая точка tl(X=200,Y=200)

// Инициализация графики

int a=DETECT,b;

initgraph(&a,&b,"F:\\BC5\\bgi");

t.show(); . / / показать точку t

cout « "\n коор-ты точки t: x=" « t.putx() « ",y=" « t.puty();
getch(); //ждать нажатия клавиши

tl.show(); //показать тоочку tl
```

```
cout « "\n координаты точки tl: x=" « tl.putx() « ",y=" «

tl.puty();

getch();

t.move(150,300); //переместить точку t(x=150,y=300)

cout « "\n новые координаты t: x=" « t.putx() « ",y=" «

t.puty(); getch();

closegraph(); //закрыть графический режим

}
```

В программе используется функция *getch(),* которая позволяет остановить выполнение программы до нажатия на клавиатуре любой клавиши. Прототип этой функции находится в файле conio.h.

Оценка	Показатели оценки
3	Ввести текст программы в среду в С++ Определение типов переменных Задание типов переменных. Определить компоненты и их свойства
4	Ввести текст и отладить программу получить результат Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма

5	Отладить программу и объяснить ее работу Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Текущий контроль №14

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита
Задание №1

Файл circle.h - описание производного класса

```
#include "f:\POS C:\PRIMER\point.cpp" // описание класса Point

class Circle: public Point //класс для определения окружности

{

// модификатор public позволяет сделать элементы x и y класса

// Point в описании класса Circle защищенными (protected)
protected: //защищенный статус доступа к элементам данных

int radius; //радиус окружности

int vis; //видимость окружности на экране

//Прототипы методов:

public: //общедоступный статус доступа

Circle(int,int,int); //конструктор

~Circle(); //деструктор

void show(); //изобразить окружность на экране

void hide(); //убрать изображение окружности
```

void move (int,int); //переместить окружность

void vary(int); //изменить размер окружности

int putradius (); // доступ к радиусу

};

Оценка	Показатели оценки
3	Ввести текст программы Определение типов переменных Задание типов переменных. Определить компоненты и их свойства
4	Ввести текст и отладить программу Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Отладить программу и обяснить ее работу Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Задание №2

Используем внешние описания методов, поместив их в отдельный файл с именем *circle.cpp:*

//Файл circle.cpp

#include <graphics.h> // прототипы функций графической библиотеки

#include "f:\POS_C\PRIMER\circle.h" // описание класса Circle

//Описания методов класса Circle:

//Конструктор

```
Circle :: Circle (int xc, int yc, int radc=0) : Point(xc,yc)
{ radius=radc; vis=l; }
//Изобразить окружность на экране
void Circle :: show(void)
{ circle (x,y,radius); } //рисование окружности
//Убрать окружность с экрана
void Circle :: hide(void)
{
unsigned int col; //объявление переменной для текущего цвета
if (vis==0) return; //окружности нет
col=getcolor(); //запоминание текущего цвета
setcolor(getbkcolor()); //выявление текущего цвета фона
circle(x,y,radius); //рисование окружности цветом фона
vis=0;
setcolor(col); //восстановление текущего цвета
}
// Переместить окружность в новое место экрана
void Circle :: move(int xn, int yn)
{
hide(); //стирание старой окружности
x=xn; y=yn; show(); } //рисование новой окружности
// Изменить размер окружности
void Circle :: vary(int d)
{
```

```
Hide(); //стирание старой окружности

radius+=d; //изменение радиуса

if(radius<0) radius=0;

show(); //рисование новой окружности

// Доступ к радиусу

int Circle :: putradius()

{ return radius; }

// Деструктор

Circle :: ~Circle()

{ hide(); } //стирание окружности
```

Оценка	Показатели оценки
3	Разработка системы тестов, охватывающих: средние значения; граничные условия входных данных; граничные условия выходных данных.
4	Разработка системы тестов, охватывающих: средние значения; граничные условия входных данных; граничные условия выходных данных. Проверка программы на разработанной системе тестов
5	Разработка системы тестов, охватывающих: средние значения; граничные условия входных данных; граничные условия выходных данных. Проверка программы на разработанной системе тестов Заключение о работоспособности алгоритма, реализованном в программе

Задание №3

Теперь можно составить программу для работы с объектами класса *Circle.*

```
#include <iostream,h>

#include <graphics.h> //прототипы графических функций

#include <conio.h> //прототип функции getch()

#include "f:\POS C\PRIMER\circle.cpp" //описание класса Circle

int main () {

// Инициализация графики

int a=DETECT,b;

initgraph(&a,&b,"F:\\BC5\\bgi");

Circle A(150,250,30); //создана невидимая окружность A

Circle B(300,100,50); //создана невидимая окружность B
cout « "\n A: x=" « A.putx() « ", y=" « A.puty() «", радиус=" « A.putradius ();

cout « "\n B: x=" « B.putx(} « ", y=" « B.puty () « ", радиус=" « B.putradius () ;

getch(); //ждать нажатия клавиши

A.show(); //показать на экране окружность A

getch(); //ждать нажатия клавиши

B.show(); //показать на экране окружность B

getch(); //ждать нажатия клавиши

Point C(300,100); //создана точка C

C.show(); // показать на экране точку C

getch(); // ждать нажатия клавиши

A.move(150,150); // переместить окружность A
cout « "\n A: x=" « A.putx() « ", y=" « A.puty() « ", радиус ="«A.putradius ();

getch();
```

```
B.vary(20); // изменить размеры окружности B
cout«"\n B: x=" « B.putx() « ", y=" « B.puty() « ", радиус=" « в. put radius () ;

getch();

A.Circle :: ~Circle{}; // уничтожение объекта A

B.Circle :: ~Circle(); // уничтожение объекта B

closegraph(); // закрыть графический режим
}
```

В результате выполнения программы вначале конструктором *Circle()* создаются два невидимых объекта: окружность *A* с координатами (150, 250) и радиусом 30 и окружность *B* с координатами (300, 100) и радиусом 50. В результате двукратной работы метода *show()* объекты *A* и *B* становятся видимыми. При явном вызове конструктора *Point()* создается объект *C* с координатами центра окружности *B.* В результате работы метода *show()* класса *Point* точка *C* становится видимой. Метод *move()* стирает старую окружность *A,* изменяет в объекте *A* значения защищенных координат объекта (150, 150) (функции, которые не принадлежат к классам *Point* и *Circle,* этого сделать не могут) и рисует окружность прежнего радиуса на новом месте экрана. Метод *vary()* стирает окружность объекта *B* с экрана (при этом объект *B* не уничтожается) и после изменения значения радиуса рисует окружность с другим радиусом.

Оценка	Показатели оценки
3	Ввести текст программы Определение типов переменных Задание типов переменных. Определить компоненты и их свойства
4	Ввести текст и отладить программу Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Отладить программу и обяснить ее работу Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Текущий контроль №15

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита
Задание №1

```
// Файл rectang.h - описание класса прямоугольник

#include <graphics.h> //прототипы функций графической библиотеки

#include "f:\POS_C\PRIMER\point.cpp" //описание класса Point

class Rectangle: public Point //класс для определения прямоугольника

{

protected //защищенный статус доступа к элементам данных

int lx; //длина по горизонтали

int ly; //длина по вертикали

// Прототипы методов:

void risrec(); //прорисовка прямоугольника

public: //общедоступный статус доступа

Rectangle(int,int,int,int) // конструктор

void show (void); //прорисовка прямоугольника

void hide(void); //убрать изображение прямоугольника с экрана

};
```

Оценка	Показатели оценки
3	Ввести текст программы в среду C++ Определение типов переменных Задание типов переменных. Определить компоненты и их свойства

4	Ввести текст программы, отладить ее Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Отладить программу и объяснить Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Задание №2

```
// Файл rectang.cpp - описания методов класса прямоугольник
#include <graphics.h> // прототипы функций графической библиотеки
#include "f:\POS_C\PRIMER\rectang.h" // описание класса Point
// Описания методов:
void Rectangle :: risrec() // прорисовка прямоугольника
{
int g=lx/2;
int v=ly/2;
line(x-g, y-v, x+g, y-v);
line(x-g, y+v, x+g, y+v);
line(x-g, y-v, x-g, y+v) ;
line(x+g, y-v, x+g, y+v);
}
```

```
// Конструктор:
Rectangle :: Rectangle{int xi, int yi, int lxi=0, int lyi=0) :
Point (xi,yi)
{
lx=lxi;
ly=lyi;
}
// Изобразить прямоугольник на экране:
void Rectangle :: show(void)
{
risrec(); // прорисовка прямоугольника
}
// Убрать изображение прямоугольника с экрана:
void Rectangle :: hide(void)
{
int b,c;
b=getbkcolor(); // запоминание текущего цвета фона
c=getcolor(); // запоминание цвета изображения
setcolor(b); // установить цвет изображения
risrec(); // нарисовать прямоугольник цветом фона
setcolor(c); // восстановление цвета изображения
}
```

Оценка	Показатели оценки

3	Ввести текст программы в среду C++ Определение типов переменных Задание типов переменных. Определить компоненты и их свойства
4	Ввести текст программы, отладить ее Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Отладить программу и объяснить Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Текущий контроль №16

Форма контроля: Практическая работа (Опрос)
Описательная часть: защита
Задание №1

Теперь можно создать класс «прямоугольник и окружность» на основании двух непосредственных

базовых классов. Описание класса Reccircle поместим в отдельный файл reccirc.h:

```
// Файл reccirc.cpp - описание класса "прямоугольник и окружность"

#include "f:\POS_C\PRIMER\rectang.cpp" // описание класса Rectangle

#include "f:\POS_C\PRIMER\circle.cpp" // описание класса Circle

// Класс "прямоугольник и окружность":

class Reccircle: public Rectangle, public Circle
```

```
{
public:
// Конструктор:
Reccircle(int xi, int yi, int ri, int lxi, int lyi) :
Rectangle(xi, yi, lxi, lyi), // Явный вызов конструкторов
Circle(xi, yi, ri) // базовых классов
{ }
// Изобразить прямоугольник и окружность на экране:
void show(void)
{
Rectangle :: show(); // изобразить прямоугольник
Circle :: show(); // изобразить окружность }
// Убрать изображение с экрана:
void hide(void)
{
Rectangle :: hide(); // убрать изображение прямоугольника
Circle :: hide(); // убрать изображение окружности
}
};
```

Оценка	Показатели оценки
3	Ввести текст программы в среду C++ Определение типов переменных Задание типов переменных. Определить компоненты и их свойства

4	Ввести текст и отладить программу Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Отладить программу и объяснить Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Задание №2

Составить программу, реализующую вычисления по следующей

формуле: s=a*b+c*k+a*c.

```
#include <iostream.h>

struct Pro

{

private:

int x, y, z;

public:

// Прототипы методов:

Pro (int, int); //конструктор

int putx(); // доступ к x

int puty(); // доступ к y

int putz(); // доступ к z
```

```
void proizv(); //произведение
~Pro(); // деструктор
};
// Описания методов:
Pro :: Pro(int xl,int yl) { x=xl; y=yl;}
int Pro :: putx() { return x;}
int Pro :: puty() ( return y;}
int Pro :: putz() { return z;}
void Pro :: proizv() {z=x*y;}
Pro :: ~Pro() { }
void main()
 {
int s,a,b,c,k;
cout « "\n Введите a,b,c и k\n";
cin » a » b » c » k;
Pro D = Pro(a,b); //создание и инициализация объекта D
Pro E(c,k); //создание и инициализация объекта E
Pro F(a,c); //создание и инициализация объекта F
D.proizv(); //получение произведения a*Ь
E.proizv(); //получение произведения c*к
F.proizv(); //получение произведения a*c
cout « "\n D.a=" « D.putx();
cout « "\t D.b=" « D.puty();
cout « "\t D.z=" « D.putz();
```

s=D.putz () +E.putz () +F.putz ();

cout « "\n s=" « s;

F.Pro :: ~Pro(); //уничтожение объекта F

E.Pro :: ~Pro(); //уничтожение объекта E

D.Pro :: ~Pro(); //уничтожение объекта D

}

Оценка	Показатели оценки
3	Ввести текст программы в среду C++ Определение типов переменных Задание типов переменных. Определить компоненты и их свойства
4	Ввести текст и отладить программу Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма
5	Отладить программу и объяснить Определение типов переменных Задание типов переменных. Определить компоненты и их свойства Реализация алгоритма на выбранном языке программирования без ошибок в логической структуре алгоритма Успешный запуск компилятора

Текущий контроль №17

Форма контроля: Индивидуальные задания (Опрос)
Описательная часть: защита
Задание №1

Создание различных тестов в среде Delphi

Оценка	Показатели оценки
3	Простой тест, по алгоритму со множеством форм
4	Тест по алгоритму с массивами вопросов и ответов
5	Сложный тест с добавлением регистрации, таймера и вывода результатов в файл

Текущий контроль №18

Форма контроля: Индивидуальные задания (Опрос)
Описательная часть: защита
Задание №1

Создание WEB - страниц с использование языка гиперссылок HTML и JAVA Script

Оценка	Показатели оценки
3	Простая WEB -страница
4	WEB - страница с хорошим дизайном, фреймами, таблицами .
5	WEB -страница с подключеним звука , анимации, видео

Задание №2

Создание WEB - страниц с использование языка гиперссылок HTML и JAVA Script

Оценка	Показатели оценки
3	Простая WEB -страница
4	WEB - страница с хорошим дизайном, фреймами, таблицами .
5	WEB -страница с подключеним звука , анимации, видео

Задание №3

Создание WEB - страниц с использование языка гиперссылок HTML и JAVA Script

Оценка	Показатели оценки
3	Простая WEB -страница
4	WEB - страница с хорошим дизайном, фреймами, таблицами .
5	WEB -страница с подключеним звука , анимации, видео