

**Контрольно-оценочные средства для проведения текущего
контроля
по УП.1 Учебной практики
(3 курс, 6 семестр 2022-2023 уч. г.)**

Текущий контроль №1

Форма контроля: Практическая работа (Сравнение с аналогом)

Описательная часть: Защита

Задание №1

Задание Создать COM-файл

title Primer1

add segment para

assume ss:add,ds:add,cs:add

org 100h

vv: jmp nach

nam1 db '123456'

nam2 db 'abcdef'

nach proc near

lea si,nam1

lea di,nam2

mov cx,6

m1:

mov al,[si]

mov ah,[di]

mov [si],ah

mov [di],al

inc si

inc di

dec cx

jnz m1

ret

nach endp

add ends

end vv

Оценка	Показатели оценки
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №2

Задание Создать COM-файл

title Primer1

add segment para

assume ss:add,ds:add,cs:add

org 100h

vv: jmp nach

nam1 db '123456'

nam2 db 'abcdef'

nach proc near

lea si,nam1

lea di,nam2

mov cx,6

m1:

mov al,[si]

mov ah,[di]

mov [si],ah

mov [di],al

inc si

inc di

dec cx

jnz m1

ret

nach endp

add ends

end vv

Оценка	Показатели оценки
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №3

Задание Создать COM-файл

title Primer1

add segment para

assume ss:add,ds:add,cs:add

org 100h

vv: jmp nach

nam1 db '123456'

nam2 db 'abcdef'

nach proc near

lea si,nam1

lea di,nam2

```

mov cx,6

m1:

mov al,[si]

mov ah,[di]

mov [si],ah

mov [di],al

inc si

inc di

dec cx

jnz m1

ret

nach endp

add ends

end vv

```

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

Задание №4

Задание Создать COM-файл

```

title Primer1

add segment para

assume ss:add,ds:add,cs:add

org 100h

vv: jmp nach

```

nam1 db '123456'

nam2 db 'abcdef'

nach proc near

lea si,nam1

lea di,nam2

mov cx,6

m1:

mov al,[si]

mov ah,[di]

mov [si],ah

mov [di],al

inc si

inc di

dec cx

jnz m1

ret

nach endp

add ends

end vv

Оценка	Показатели оценки
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №5
Задание Создать COM-файл

title Primer1

```

add segment para

assume ss:add,ds:add,cs:add

org 100h

vv: jmp nach

nam1 db '123456'

nam2 db 'abcdef'

nach proc near

lea si,nam1

lea di,nam2

mov cx,6

m1:

mov al,[si]

mov ah,[di]

mov [si],ah

mov [di],al

inc si

inc di

dec cx

jnz m1

ret

nach endp

add ends

end vv

```

Оценка	Показатели оценки

3	Разработка системы тестов, охватывающих средние значения и граничные условия входных и выходных данных.
4	Проверка программы на разработанной системе тестов
5	Дать заключение о работоспособности алгоритма, реализованном в программе

Текущий контроль №2

Форма контроля: Практическая работа (Сравнение с аналогом)

Описательная часть: Защита

Задание №1

Задание Написать программу на языке Ассемблер.

1.Ввести строку символьных данных ,задавая буфер равный 18 байт.

Подсчитать в этой строке количество символов "i" .

~~Выдать подсчитанное количество символов.~~

Оценка	Показатели оценки
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №2

Задание Написать программу на языке Ассемблер.

1.Ввести строку символьных данных ,задавая буфер равный 18 байт.

Подсчитать в этой строке количество символов "i" .

~~Выдать подсчитанное количество символов.~~

Оценка	Показатели оценки
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №3

Задание Написать программу на языке Ассемблер.

1.Ввести строку символьных данных ,задавая буфер равный 18 байт.

Подсчитать в этой строке количество символов "i" .

~~Выдать подсчитанное количество символов.~~

Оценка	Показатели оценки
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №4

Задание Написать программу на языке Ассемблер.

1.Ввести строку символьных данных ,задавая буфер равный 18 байт.

Подсчитать в этой строке количество символов "i" .

Выдать подсчитанное количество символов.

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

Задание №5

Задание Написать программу на языке Ассемблер.

1.Ввести строку символьных данных ,задавая буфер равный 18 байт.

Подсчитать в этой строке количество символов "i" .

Выдать подсчитанное количество символов.

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

Текущий контроль №3

Форма контроля: Практическая работа (Сравнение с аналогом)

Описательная часть: Защита

Задание №1

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

Написать программу, которая вычисляет сумму первых членов ряда: 1,3,5,7,.....Количество

суммируемых

членов ряда задается во время работы программы.

Оценка	Показатели оценки
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №2

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

Написать программу, которая вычисляет сумму первых членов ряда: 1,3,5,7,.....Количество

суммируемых

членов ряда задается во время работы программы.

Оценка	Показатели оценки
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №3

1. Написать программу, которая выводит таблицу квадратов десяти первых положительных чисел
Написать программу, которая вычисляет сумму первых членов ряда: 1,3,5,7,...Количество

суммируемых

членов ряда задается во время работы программы.

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

Задание №4

1. Написать программу, которая выводит таблицу квадратов десяти первых положительных чисел
Написать программу, которая вычисляет сумму первых членов ряда: 1,3,5,7,...Количество

суммируемых

членов ряда задается во время работы программы.

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

Текущий контроль №4

Форма контроля: Практическая работа (Сравнение с аналогом)

Описательная часть: Защита

Задание №1

Написать программы на языке C++

1. Дана последовательность целых чисел $a_1, a_2, \dots, a_n$. Выяснить, какое число встречается раньше

—

положительное или отрицательное. Оценка	Показатели оценки
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №2

Написать программы на языке C++

1. Дана последовательность целых чисел $a_1, a_2, \dots, a_n$. Выяснить, какое число встречается раньше

—

положительное или отрицательное. Оценка	Показатели оценки
3	Разработка системы тестов, охватывающих средние значения и граничные условия входных и выходных данных.
4	Проверка программы на разработанной системе тестов
5	Дать заключение о работоспособности алгоритма, реализованном в программе

Задание №3

Написать программы на языке C++

1. Дана последовательность целых чисел $a_1, a_2, \dots, a_n$. Выяснить, какое число встречается раньше

—

положительное или отрицательное. Оценка	Показатели оценки
3	Разработка системы тестов, охватывающих средние значения и граничные условия входных и выходных данных.
4	Проверка программы на разработанной системе тестов
5	Дать заключение о работоспособности алгоритма, реализованном в программе

Текущий контроль №5

Форма контроля: Практическая работа (Сравнение с аналогом)

Описательная часть: Защиты

Задание №1

Задание 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

```

Задание 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

```

Оценка	Показатели оценки
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы
5	Успешный запуск компилятора

Задание №2

Задание 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() const //доступ к x
int puty(); //доступ к y
void show(); //изобразить точку на экране

void move (int,int); //переместить точку
private: //собственный статус доступа
void hide(); //убрать изображение точки
};
#endif

```

Задание 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
```

Оценка	Показатели оценки
3	Анализ программы на соответствие блок-схеме
4	Отладка семантики и логической структуры программы

5	Успешный запуск компилятора
---	-----------------------------