**Структуры и объединения**

**Цель лабораторной работы**: освоить структуры и объединения с помощью языка программирования C.

**Задача 1**

**Постановка задачи**: 

**Программный код:**

#include <stdio.h>

#include <stdlib.h>

typedef struct str

{

int(\*func)();

} myStr;

void proc()

{

printf("Hello, world!");

}

int main() {

myStr obj;

obj.func = &proc;

obj.func();

printf("\n");

system("pause");

return 0;

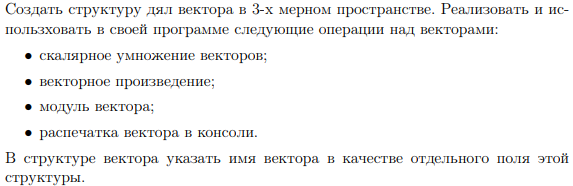
}

**Вывод консоли:**



**Задача 2**

**Постановка задачи**:



**Программный код:**

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

typedef struct vector

{

int x;

int y;

int z;

char name;

} vector;

int main()

{

vector obj1 = { 1, 1, 1, {'obj1'} };

vector obj2 = { 1, 1, 1, {'obj2'} };

printf("Vector %c:\n %i, %i, %i\n", obj1.name, obj1.x, obj1.y, obj1.z);

printf("Vector %c:\n %i, %i, %i\n", obj2.name, obj2.x, obj2.y, obj2.z);

printf("Scalar product of vektors:\n %d\n", obj1.x\*obj2.x + obj1.y\*obj2.y + obj1.z\*obj2.z);

int x = obj1.y \* obj2.z - obj1.z \* obj2.y;

int y = -(obj1.x \* obj2.z - obj1.z \* obj2.x);

int z = obj1.x \* obj2.y - obj1.y \* obj2.x;

printf("Vector product of vectors:\n %i, %i, %i\n", x, y, z);

printf("Module of vektor %c:\n %lf\n", obj1.name, sqrt(obj1.x\*obj1.x + obj1.y\*obj1.y + obj1.z\*obj1.z));

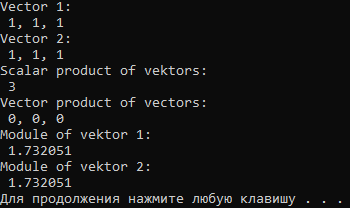
printf("Module of vektor %c:\n %lf\n", obj2.name, sqrt(obj2.x\*obj2.x + obj2.y\*obj2.y + obj2.z\*obj2.z));

system("pause");

return 0;

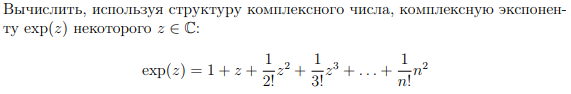
}

**Вывод консоли:**



**Задача 3**

**Постановка задачи**:



**Программный код:**

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

typedef struct c

{

float x;

float y;

} setOfC;

int main(){

int n, i;

int fac = 1;

float corner;

setOfC z;

printf("Enter x, y and n:\n ");

scanf("%fac", &z.x);

scanf("%fac", &z.y);

scanf("%i", &n);

corner = (float)(z.y / z.x);

z.x = 1;

z.y = 1;

for (i = 1; i < n; i++){

if (i != 0)

fac = fac \* i;

z.x += (float)(1 / fac \* cos(i \* corner));

z.y += (float)(1 / fac \* sin(i \* corner));

}

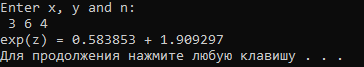
printf("exp(z) = %f + %f\n", z.x, z.y);

system("pause");

return 0;

}

**Вывод консоли:**



**Задача 4**

**Постановка задачи**:



**Программный код:**

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

typedef struct Struct

{

char list;

struct Struct \*next;

struct Struct \*pre;

} str;

int main(){

int i, size;

str \*first = malloc(sizeof(str));

first->list = '1';

first->pre = NULL;

str \*x, \*y;

x = first;

for (i = 1; i < 6; i++){

y = malloc(sizeof(str));

y->list = x->list + 1;

x->next = y;

y->pre = x;

x = y;

}

x->next = 0;

x = first;

while (x != 0){

printf("%c ", x->list);

x = x->next;

}

printf("\n");

x = first->next->next->next->next->next;

while (x != 0){

printf("%c ", x->list);

x = x->pre;

}

printf("\n");

free(first->next->next->next->next->next);

free(first->next->next->next->next);

free(first->next->next->next);

free(first->next->next);

free(first->next);

free(first);

system("pause");

return 0;

}

**Вывод консоли:**

