**Subject in Spring Session 2022:**

|  |  |
| --- | --- |
| **1)Object Oriented Programming:**  Lectures  Practice Lessons (Laboratory Lessons) | **Tuesday:**  **1030-1200 303 U2**  **1230-1400 Department DSM, room #11** |
| **2)Computing Architecture:**  Practice Lessons (Laboratory Lessons) | **Wednesday:**  **1430-1600Department DSM, room #4** |

**Teacher:**

Surname: **Rozova**

Name: **Lyudmila Viktorovna**

**Associate Professor, Ph.D**

**Email:** [**Lyudmyla.rozova@khpi.edu.ua**](mailto:Lyudmyla.rozova@khpi.edu.ua)

**Telegram: +380507397805**

**The materials of subject:** [**https://github.com/LRozova/OOP\_eng**](https://github.com/LRozova/OOP_eng)

**Introduction. Programming.**

C++ programming language.

**The Review of Previous Topics:**

* types;
* arrays (one-dimensional, two-dimensional)
* pointers, dynamic memory;
* functions.

Variables of different types:

|  |  |
| --- | --- |
| int a,b;  float d,e;  double f;  char c;  a=2;  d=1.03;  c=’b’; | int x[10];  int A[3][4];  float C[4][5];  ….  x[0]=3;  A[0][1]=3.1; |

**Functions**

**type імя\_функції (the list of arguments)**  // the title of function

**{**

**Function’s body;**

**}**

**type** – the type of the value returned by the function

**for example:**

int suma(int a, int b)

{

return a+b;

}

//---------------------------

void input(x[10])

{ cout<<”Enter array x”<<endl;

for(i=0;i<10;i++)

cin>>x[i];

}

**Example 1:**

#include <iostream>

#include <cstdlib>

#include <ctime>

using namespace std;

void printMas(int [10]);

void input(int A[10]);

int main()

{

int X[10];

input(X);

printMas(X);

return 0;

}

void input(int A[10])

{

srand(time(0));

for(int i = 0; i < 10; i++)

{

A[i] = rand()%20-1;

}

}

void printMas(int A[10])

{

for(int i = 0; i < 10; i++) {

cout << A[i] << " ";

}

cout << endl;

}

**Task 1:** workout for the brain!

Add to the previous program:

* function **sort**  – to sort in ascending order of array elements;
* function **sum\_pos –** to find the sum of positive array elements

**Solution for the task 1:**

#include <iostream>

#include <cstdlib>

#include <ctime>

using namespace std;

void printMas(int [10]);

void input(int A[10]);

void sort(int A[10]);

int sum\_pos(int A[10]);

int main()

{

  int X[10];

  input(X);

  cout<<"Array before sorting"<<endl;

  printMas(X);

  sort(X);

  cout<<"Array after sorting"<<endl;

  printMas(X);

  cout<<"Sum\_pos="<<sum\_pos(X)<<endl;

  return 0;

}

void input(int A[10])

{

  srand(time(0));

  for(int i = 0; i < 10; i++)

{

  A[i] = rand()%20-1;

}

}

void printMas(int A[10])

{

  for(int i = 0; i < 10; i++) {

  cout << A[i] << " ";

}

  cout << endl;

}

void sort(int A[10])

{

    int tmp,i,j;

    for(i = 0; i < 10; i++)

    for(j = 0; j < 10-i-1; j++)

     if(A[j] > A[j+1]) {

       tmp = A[j];

       A[j] = A[j+1];

       A[j+1] = tmp;

    }

}

int sum\_pos(int A[10])

{

    int S,i;

    for(i=0;i<10;i++)

        if(A[i]>0)

         S+=A[i];

    return S;

}

Pointers

int \*p;

p=&a;

\*p=4;

cout<<\*p;

Dynamic memory.

Allocation and release of dynamic memory.

**int \*p;**

**pointer’s name = new type**

p=new int;

**delete pointer’s name**

delete p;

For one-dimensional array:

**pointer’s name = new type [size];**

p=new int[size];

**delete [size] pointer’s name**

delete [size] p; or delete p;

**Task 2:**

Change program from the task 1, using **dynamic array**

**Example 2.**

#include <iostream>

#include <cstdlib>

#include <ctime>

using namespace std;

int main()

{

int \*\*A;/ / declare a double pointer

int n=3,m=3,i,j;

//dynamic memory allocation in two steps

//for the array of pointers to arrays of columns

A = new int\*[n];

for(i=0;i<n;i++)

A[i] = new int[m];

//checking dynamic memory allocation

if(!A)

{cout<<"Not memory \n";}

srand(time(0));// resetting the function to generate random numbers

//random filling of array elements

//in a double cycle

for(i=0;i<n;i++)

for(j=0;j<m;j++)

{ //random filling of array elements in the range from -10 to 20

A[i][j]= rand()%20-10;

//output of random elements

cout<<"\nA["<<i<<","<<j<<"]="<<A[i][j];}

//dynamic memory release

for(i=0;i<n;i++)

delete A[i];

delete A;

return 0;

}

**Task 3:**

**Change the program from example 2.**

**Add:**

* function **fill –** for random filling of array;
* function **print –** to print array as a matrix;
* function **pos\_el\_row** – which calculate and print the sum of positive elements in each row.

**Task 4:**

**Change the program from task 3.**

* Instead of function **fill** add function **createMas**, which allocate dynamic memory for array, and read the values of array from the text file 1
* function **print –** to print array as a matrix to the file 2;