**ՀԱՅԱՍՏԱՆԻ ԱԶԳԱՅԻՆ ՊՈԼԻՏԵԽՆԻԿԱԿԱՆ**

**ՀԱՄԱԼՍԱՐԱՆ**

**ՏՀՏԷ ինստիտուտ**

**ՏԵՂԵԿԱՏՎՈՒԹՅԱՆ ԱՆՎՏԱՆԳՈՒԹՅԱՆ ԵՎ ԾՐԱԳՐԱՅԻՆ ԱՊԱՀՈՎՄԱՆ ԱՄԲԻՈՆ**

**Առարկա՝** ԾՀՊ

**Խումբ՝** ՏՏ019-3

**Ուսանողուհի՝** Ալինա Ավետիսյան

**Դասախոս՝** Ա․ Մարտիրոսյան

1․Խնդրի դրվածքը։ Գրել ֆունկցիա,որը որպես արգումենտ կստանա երկու թիվ,և կվերադարձնի այդ թվերի ամենամեծ ընդհանուր բաժանարարը,և ամենափոքր ընդհանուր բազմապատիկը։

#include <iostream>

using namespace std;

int gcd(int a, int b)

{

while (a > 0 && b > 0)

{

if (a > b)

a = a % b;

else

b = b % a;

}

if (a == 0)

return b;

else

return a;

}

int lcm(int a, int b, int gcd)

{

int lcm;

lcm = (a \* b) / gcd;

return lcm;

}

int gcd\_lcm()

{

int a, b, sel;

while (1) {

cout << "Choose 1 or 2 or 0 to exit" << endl;

cin >> sel;

if (!sel)

return 0;

else {

cout << "Input 2 numbers" << endl;

cin >> a >> b;

cout << "Output" << endl;

if (sel == 1) {

cout << gcd(a, b);

}

else if (sel == 2)

{

cout << lcm(a, b, gcd(a, b));

}

cout << endl;

}

}

}

2. Խնդրի դրվածքը։ n տարր պարունակող զանգվածի տարրերը 1-ից n+1 չկրկնվող թվերն են։ Գրել ֆունկցիա, որը կտպի այն թիվը, որը պակասում է այդ զանգվածում։

#include <iostream>

using namespace std;

void arr\_srch()

{

int arr[16] = { 9,13,3,16,1,6,14,8,5,10,7,4,11,0,15,2 };

int count = 0;

for (int i = 0; i <= 17; i++)

{

for (int j = 0; j <= 16; j++)

{

if (i == arr[j])

count = 1;

}

if (count == 0)

{

cout << i;

break;

}

count = 0;

}

}

3․ Խնդրի դրվածքը։ Գաղտնագրում շաբլոնի մեթոդով։ Այն տողը, որը պետք է գաղտնագրել, համեմատում ենք encode\_string –ի հետ, եթե այդ տարրը կա encode\_string –ում, ապա decode\_string –ի համապատասխան ինդեքսով տարրն ենք տպում, իսկ եթե չկա, ապա տպում ենք հենց այդ տարրը։

#include <iostream>

using namespace std;

string real = "adtahyqi12sy85ebgdarhvdcvcdvcdhhgg";

string unreal = "xxfmgoph35jkqwlmnksdgddfghjkfghtjg";

string encoder(string s){

int j;

int i = 0;

while(s[i]){

j = 0;

while(unreal[j]){

if(s[i] == unreal[j]){

s[i] = real[j];

break;

}

j++;

}

i++;

}

return s;

}

string dcoder(string s){

int j;

int i = 0;

while(s[i]){

j = 0;

while(real[j]){

if(s[i] == real[j]){

s[i] = unreal[j];

break;

}

j++;

}

i++;

}

return s;

}

int main(){

string s = "this year";

string realresult = dcoder(s);

string unrealresult = encoder(realresult);

cout << realresult << endl << unrealresult;

}

4․ Խնդրի դրվածքը։ Գաղտնագրում տողեր/սյուներ մեթոդով։

#include <iostream>

using namespace std;

void row\_col\_encrypt()

{

char vec[] = "012345678abcdef";

int key = 3;

char arr[5][3];

int cnt=0;

for (int i = 0; i < 5; i++)

{

for (int j = 0; j < 3; j++)

{

arr[i][j] = vec[cnt];

cnt++;

cout << arr[i][j] << " ";

}

cout << endl;

}

cout << endl;

cnt = 0;

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 5; j++)

{

vec[cnt] = arr[j][i];

cout << vec[cnt] << " ";

cnt++;

}

}

}

5 ․ Խնդրի դրվածքը։ Տվյալների սեղմում RLE մեթոդով։ Մեթոդը կայանում է հետևյալում․ սկզբում տպում ենք առաջին տառը, այնուհետև հաշվում ենք, թե այդ տառը քանի անգամ է գրված մինչև իրենից տարբեր մյուս տառը, և տպում քանակը։ Հետո տպում հաջորդ տառը և նորից տպում նրա քանակը, մինչև մյուս տառը, և այդպես շարունակ, մինչև տողի ավարտը։

#include <stdio.h>

int rle()

{

char str[11] = "AAABCCGTTT\0";

char str1[5] = "\0";

int i = 0;

int j = 0;

int count = 0;

int k = 0;

printf("%s\n", str);

while (str[i])

{

j = i;

count = 0;

while (str[j] == str[i])

{

count++;

j++;

}

str1[k] = str[i];

k++;

if (count > 1) {

str1[k] = count + 48;

k++;

}

i += count;

}

str1[k] = '\0';

printf("%s\n", str1);

}