T.C.
İSTANBUL ÜNİVERSİTESİ
Mühendislik Fakültesi
Bilgisayar Mühendisliği Bölümü

Dersin Kodu: BIMU1052	**Dersin Adı: INTRODUCTION TO ALGORITHMS**
Dersin Öğretim Üyesi: Yrd.Doç.Dr. Özgür Can TURNA	**Sınav Türü: Bütünleme**
Sınav Tarihi ve Süresi: 15.01.2018 (75dk)	**Öğrenci No:**
Öğrenci Ad - Soyad:	**İmzası:**

1. (20p) Write the output of these code segments.

a.

```
int x = 30, y = 4, z = 5;
x = x / y + z + x;
cout << "X:" << x;
```

```
X: 42
```

b.

```
int x = 1, y = 2;
if ( x = y)
       x += ++y;
cout << "X:" << x << " & Y:" << y;
```

```
X: 5  & Y : 3
```

c.

```
int a = 5, b = 2;
float c = a * 3.0 / b;
cout << "C:" << c;
```

```
C: 7.5
```

d.

```
int a = 3, b = 4; int *c = &a;
*c *= a + b++;
cout << "A:" << a << " & B:" << b;
```

```
A: 21 & B: 5
```

2. (10p) Write the output of this program below.

```
#include <iostream>
using namespace std;
int main()
{
       char *ptr = NULL, name[50] = "1234567890*?";
       int i, j, n = 0;
       for (i = 0; name[i] != '\0'; i++){
              n = i;
       }

       for (i = 0; i <= n ; i++)
       {
              if (i % 2 == 0) {
                     j = name[i];
                     name[i] = name[n - i];
                     name[n - i] = (char)j;
              }
              if (i % 5 == 0)
              {
                     j = name[i];
                     name[i] = name[n - i];
                     name[n - i] = (char)j;
              }
       }
       cout << name;
}
```

```
1209867543*?
```

3. (30) Write a program to calculate the formula below. n and k will be taken from user.

$$ {}_nC_k \equiv \binom{n}{k} \equiv \frac{n!}{(n-k)!\,k!}, $$

```
#include <iostream>
using namespace std;

float fact(float a)
{
	float factor = 1;
	for (float i = a; i > 1; i--)
	{
		factor *= i;
	}
	return factor;
}

int main()
{
	int n, k;
	cout << "Enter n and k for C(n,k):" << endl;
	cin >> n;
	cin >> k;
	float result = fact(n) / (fact(n - k)*fact(k));
	cout << "C(n,k):" << result;
	cin >> n;
	return 0;
}
```

4. (20p)You have a struct definition like below. You will implement findHighest3 function. It finds highest 3 agno in the org array and prints the student details in number order.

```
struct student {
	char name[50];
	long number;
	float agno;
};
typedef struct student STD;
STD ogr[100];
void findHighest3();  //Implement this function
```

```
#include <iostream>
using namespace std;

struct student {
	char name[50];
	long number;
	float agno;
};
typedef struct student STD;
STD ogr[100];
void findHighest3();  //Implement this function

void findHighest3() {
	int max_index1 = -1, max_index2 = -1, max_index3 = -1;
	float max1 = 0, max2 = 0, max3 = 0;

	for(int i = 0; i < 100 ; i++)
	{	//max1 > max2 > max3
		if (ogr[i].agno > max3) {
			max3 = ogr[i].agno;
			max_index3 = i;
			if (max3 > max2) {
				max2 = max3;
				max_index2 = max_index3;
				if (max2 > max1)
				{
					max1 = max2;
					max_index1 = max_index2;
				}
			}
		}
	}
    STD *maxRated[3] = { &ogr[max_index1] , &ogr[max_index2] , &ogr[max_index3] } , *tempAdr =
NULL;
	//Order selected records with sorting alg.
	for (int i = 0; i < 3; i++)
	{
		for (int j = i + 1; j < 3; j++) {
			if (maxRated[i]->number > maxRated[j]->number)
			{
				tempAdr = maxRated[i];
				maxRated[i] = maxRated[j];
				maxRated[j] = tempAdr;
			}
		}
	}
	for (int i = 0; i < 3; i++)
	{	//Print RESULTS
		cout << "Student: Name:" << maxRated[i]->name << " Number:" << maxRated[i]->number
<< " Agno:" << maxRated[i]->agno << endl;
	}
}
```

5. (10p) Write the output of this program below.

```
#include <iostream>
using namespace std;
int main()
{
    int X[] = { 10, 25, 30, 55, 110 };
    int *p = X;
    while ( *p < 110)
    {
        if ( *p % 3 != 0)
            *p = *p + 1;
        else
            *p = *p + 2;
        p++;
    }
    for (int I = 4; I >= 1; I--)
    {
        cout << X[I] << "*";
        if (I % 3 == 0)
            cout << endl;
    }
    cout << X[0] * 3 << endl;
}
```

```
110*56*
32*26*33
```

6. (10p) Complete the code below to see the output on the right.

```
void Position(int &C1, int C2 = 3)
{
    C2 += 1;
    C1 += C2;
}

int main()
{
    int P1 = 20, P2 = 4;
    Position(P1);
    cout << P1 << ", " << P2 << endl;
    Position(P2, P1);
    cout << P1 << ", " << P2 << endl;
}
```

```
24, 4
24, 29
```