



HASAN FERDİ TURGUTLU TEKNOLOJİ FAKÜLTESİ

YAZILIM MÜHENDİSLİĞİ

ÖDEV – 1

Romalı Asker Problemi

YAZILIM SINAMA

Hazırlayanlar:

Kadir Sefa ÜNAL – 132805008

Kadir MUTLU - 132805013

İçindekiler

1. Amaç	3
2. Girdiler	3
3. Mantığı	3
4. Program Kodu	3
5. Ekran Çıktısı	3

1. Amaç

Roma Askeri Problemi orijinal adıyla Josephus Probleminin çözümü için algoritma geliştirmek ve yazılımın değişik durumlarda verdiği sonuçları karşılaştırarak test işlemine tabi tutmak amaçlanmıştır.

2. Girdiler

- Asker Sayısı: Kullanıcıdan alınan ölümü bekleyen asker sayısı.
- Adım Sayısı: Kullanıcı tarafından kaç adımda işlem yapılacağı belirlenir.

3. Mantığı

Problem Josephus'un Yahudi Roma Savaşı'ndaki komutanlık günlerine dayanıyor. Yahudilerin direnişi Roma güçlerince bastırılmış ve Josephus ve 40 askeri kıyımdan kaçarak gizli bir yere saklanmışlardı. Askerler Romalılar'ın eline düşüp tutsak olmaktansa, intihar etmenin en onurlu davranış olacağını düşünüyorlardı. Josephus'un önerisi dairesel şekilde dizilerek bir adım belirlenir ve o adımdaki asker bir önceki asker tarafından öldürülür ve döngü son 1 kişi kalana kadar devam eder.

Kullandığımız algoritmada asker sayısı ve adım sayısı alınarak son 2 kişi kalana kadar askerler birbirini öldürür.

4. Program Kodu

```
static void Main(string[] args)
{
    int kSayisi = 0, aSayisi = 0;

    Console.Write("Kişi sayısını girin: ");
    string kisiSayisi = Console.ReadLine();
    if (SayiMi.Kontrol(kisiSayisi))
    {
        kSayisi = Convert.ToInt32(kisiSayisi);

        Console.Write("Adım sayısını girin: ");
        string adimSayisi = Console.ReadLine();
        if (SayiMi.Kontrol(adimSayisi))
        {
            aSayisi = Convert.ToInt32(adimSayisi);
            if (kSayisi > 2 && aSayisi > 0)
            {
                int[] kisiler = new int[kSayisi];
                for (int i = 0; i < kSayisi; i++)
                    kisiler[i] = 1;
            }
        }
    }
}
```

```

int oluSayisi = 0, j = -1, k = 0;
while (oluSayisi != (kSayisi - 2))
{
    while (k != aSayisi)
    {
        j++;
        if (j == kSayisi)
            j = 0;
        if (kisiler[j] == 1)
            k++;
    }
    kisiler[j] = 0;
    oluSayisi++;
    k = 0;
    for (int i = 0; i < kSayisi; i++)
    {
        Console.Write(kisiler[i] + " ");
    }
    Console.WriteLine();
}

for (int i = 0; i < kSayisi; i++)
{
    if (kisiler[i] == 1)
        Console.WriteLine(i + 1);
}
}
else
    Console.WriteLine("Kişi sayısı 2'den fazla olmalı yada adım
sayısı 0'dan fazla olmalı!");
}
else
    Console.WriteLine("Adım sayısı hatalı!");
}
else
    Console.WriteLine("Kisi sayısı hatalı!");
Console.ReadKey();
}

static class SayiMi
{
    public static bool Kontrol(string ifade)
    {
        foreach (char chr in ifade)
        {
            if (!Char.IsNumber(chr)) return false;
        }
        return true;
    }
}

```

5. Ekran Çıktısı

```
file:///C:/Users/Kadir/Desktop/YazilimSinama/RomaliAskerProblemi/Ro...
Kişi sayısını girin: 10
Adım sayısını girin: 3
1 1 0 1 1 1 1 1 1 1
1 1 0 1 1 0 1 1 1 1
1 1 0 1 1 0 1 1 0 1
1 0 0 1 1 0 1 1 0 1
1 0 0 1 1 0 0 1 0 1
0 0 0 1 1 0 0 1 0 1
0 0 0 1 1 0 0 0 0 1
0 0 0 1 0 0 0 0 0 1
4
10
```

6. Test Senaryoları

- Asker sayısı girdisinin sayı olup olmadığı kontrolü başarılı.
- Adım sayısı girdisinin sayı olup olmadığı kontrolü başarılı.
- Asker sayısının 2'den fazla olup olmama kontrolü başarılı.
- Adım sayının 0'dan büyük olup olmama kontrolü başarılı.
- Son kalan 2 askerin gösterilme durumu başarılı.