



YTÜ Skylab Kulübü
Algorithm Games Challenge
Etap #1 Çözümleri

MATEMATİK BÖLÜMÜ

Cevap #1 (2 Puan)

MJQQT BTWQI → HELLO WORLD

Deşifre işlemi 5 sola kaydırılarak yapılır.

Cevap #2 (5 Puan)

13 hafta. Haftalar boyunca gerçekleştirilecek işler şöyle gösterilebilir:

1 hafta: 2'şer blokluk 500 yığın;

2 hafta: 4'er blokluk 250 yığın;

3 hafta: 8'er blokluk 125 yığın;

4 hafta: 16'şar blokluk 62 yığın ve 8 blokluk 1 yığın;

5 hafta: 32'şer blokluk 31 yığın ve 8 blokluk 1 yığın;

6 hafta: 64'er blokluk 15 yığın ve 40 blokluk 1 yığın;

7 hafta: 128'er blokluk 7 yığın ve 104 blokluk 1 yığın;

9 hafta: 256'şar blokluk 3 yığın (bir haftalık gecikmeyi anımsayın) ve 232 blokluk 1 yığın;

11 hafta: 512 ve 488 blokluk iki yığın;

13 hafta: 1 kilometre yüksekliğinde bir kule.

Önemli olan hızlı bir mikroişlemci tasarlamak değildir, işin püf noktası hızlı bir sistem geliştirebilmekte yatar. (Seymour Cray)

*1.ve 2.soru için düzenlenmiş yazıyı okumayı unutmayınız..

Cevap #3 (3 Puan)

Casusların tümü de doğruyu söylüyor olsaydı birbirini tanıyanların sayısının bir tek sayı olması gerekirdi. Her buluşma iki kişinin tanışmasına yol açtığından bu olanaksızdır.

Aşağıda casusların verdiği ifadelere göre örnek çözüm belirtilmiştir.

	<u>A</u>	<u>B</u>	<u>C</u>	<u>D</u>	<u>E</u>	<u>F</u>	<u>G</u>
A->6	AB	BA	CA	DA	EA	FA	GA
B->5	AC	BC	CB	DB	EB	FB	(?)GC
C->4	AD	BD	CD(doldu)	DC			
D->3	AE	BE(doldu)	CG(?)				
E->2	AF	BF(doldu)					
F->2	AG(doldu)						
G->1							

ALGORİTMA BÖLÜMÜ

Cevap #4 (4 Puan)

```
#include<stdio.h>
int main(){
    int i=0,j,k=7,n,son;

    printf("Enter a number:");
    scanf("%d",&n);
    int temp=n;
    if(n==0){
        i=1;
        k=k*10;
        son=k+temp;
    }
```

```

    }
    else if(n<0){
        n=n*-1;
        temp=temp*-1;
        while(n>0){
            n=n/10;
            i++;
        }
        for(j=0;j<i;j++){
            k=k*10;
        }
        son=k+temp;
        son=son*-1;
    }
    else if (n>0){
        while(n>0){
            n=n/10;
            i++;
        }
        for(j=0;j<i;j++){
            k=k*10;
        }
        son=k+temp;
    }
    printf("new n:%d",son);
    return 0;
}

```

Cevap #5 (6 Puan)

```
#include <stdio.h>
```

```

void writeSum() {
    int i,j;
    for (i=0; i<=50; i++)
        for (j=0; j<=50; j++) {
            printf(" if (sayi1 == %d && islem == '+' && sayi2 == %d ){\\n    printf(\"\\\"%d+%d
= %d\\\"");\\n }\\n",i,j,i,j,i+j);
        }
    return;
}

void writeSub() {
    int i,j;
    for (i=0; i<=50; i++)
        for (j=0; j<=50; j++) {
            printf(" if (sayi1 == %d && islem == '-' && sayi2 == %d ){\\n    printf(\"\\\"%d-%d
= %d\\\"");\\n }\\n",i,j,i,j,i-j);
        }
    return;
}

```

```

void writeDiv() {
    int i,j;
    for (i=0; i<=50; i++)
        for (j=0; j<=50; j++) {
            if (j!=0) {
                printf(" if (sayi1 == %d && islem == '/' && sayi2 == %d ){\n    printf(\"\"%d/%d
= %.2f\n\" );\n }",i,j,i,j,(float)i/j);
            } else {
                printf(" if (sayi1 == %d && islem == '/' && sayi2 == %d ){\n    printf(\"\"%d/%d =
TANIMSIZ\n\" );\n }",i,j,i,j);
            }
        }
    return;
}

void writeMul() {
    int i,j;
    for (i=0; i<=50; i++)
        for (j=0; j<=50; j++) {
            printf(" if (sayi1 == %d && islem == '*' && sayi2 == %d ){\n    printf(\"\"%d*%d
= %d\n\" );\n }",i,j,i,j,i*j);
        }
    return;
}

int main()
{
    printf("#include <stdio.h>\n");
    printf("int main()\n");
    printf("{\n");
    printf(" int sayi1, sayi2;\n char islem;\n");
    printf(" scanf(\"\"%d %%c %%d\", &sayi1, &islem, &sayi2);\n\n");

    writeSum();
    writeSub();
    writeDiv();
    writeMul();

    printf("\n return 0;\n");
    printf("}\n");

    return 0;
}

```

Soru #1 e ilişkin bir yazı...

Bu soruda Sezar şifrelemesi kullanılmıştır. Bilmeyenler için kriptolojide bebek adımları diyebiliriz.

Kriptografi, bilgi gizliliğini sağlamak için kullanılan matematiksel teknikler bütünüdür ve kriptolojinin alt dalıdır. Daha ayrıntılı araştırmayı size bırakıyoruz.

Eğer kriptoloji alanını şimdilerde keşfediyorsanız “The Imitation Game” filmi sizin için güzel bir başlangıç olabilir. Film izlerken bilgisayar biliminin doğuşuna tanıklık edeceksiniz..

Soru 2 ye ilişkin bir yazı...

Önemli olan hızlı bir mikroişlemci tasarlamak değildir, işin püf noktası hızlı bir sistem geliştirebilmekte yatar. (Seymour Cray)

Paralel hesaplama, ya da **Koşut hesaplama**, aynı görevin (parçalara bölünmüş ve uyarlanmış), sonuçları daha hızlı elde etmek için çoklu işlemcilerin eş zamanlı olarak işletilmesidir. Bu fikir, problemlerin çözümünün ufak görev parçalarına bölünmesi ve bunların eş zamanlı olarak koordine edilmesine dayanır. Paralel hesaplama ile performans artar, büyük sorunlar daha az sürede çözülür ve bilimdeki gelişmeler paralel hesaplamaya gereksinim duyar.



Hava durumu raporlarının hazırlanması, atom bombalarının geliştirilmesi, nükleer çalışmaların ve hesaplamaların yapılması, kompleks şifrelerin kırılması... Bu ve aklınıza gelebilecek en karmaşık işleri gerçekleştirebilen sistemlere Süper Bilgisayarlar diyoruz. Süper bilgisayarlar yüzlerce paralel işlemciden meydana geliyor. Günümüzdeki süper bilgisayarlar saniyede trilyonlarca işlemi yürütebilmekteler. İlk süper bilgisayar ne IBM ne de HP tarafından üretilmiştir. Şaşırtıcı gelse de ilk süper bilgisayar ABD 'de *Control Data Corporation* adlı şirket tarafından CDC 6600 modeli ile üretilmiştir. Bu ilk süper bilgisayarı insanlığın hizmetine sunan isim ise Seymour Cray 'dır. Cray daha sonra Cray Research adıyla kendi firması üzerinden yeni süper bilgisayarlar geliştirmeye devam etmiştir.

Soru 5 e ilişkin bir yazı...

Program yazan programlar, farklı programların kaynak kodlarını inceleyerek kendine ekleyen ve kendini geliştiren, yeri geldiğinde bir istenilen görevi yapmak için kod yazan programlar geleceğimizi şekillendirmesi kaçınılmaz üstelik zamanla bir çok yazılım mühendisini işsiz bırakacağı söyleniyor. Aylarca onlarca yazılımcı çalışarak üretilen projeleri saniyeler içerisinde üretip hizmete sunulabileceğini düşünmek hayret verici. Bu soruyu eklemek deki temel gaye ufak da olsa bu konu üzerine bir kıvılcım yaratabilmek.

<https://futurism.com/4-our-computers-are-learning-how-to-code-themselves>