

No	Ad - Soyad	İmza
CRN		

S1	S2	S3	S4	Toplam
----	----	----	----	--------

Süre: 100 dakika

Bilgisayar İşletim Sistemleri - Arasınay I

Soru 1: (40 puan) Aşağıdakileri 2-3 cümle ile yanıtlayınız.

a) İşletim sisteminin iki ana görevi:
b) Proses bileşenleri:
c) Zaman paylaşımli çalışma:
d) “Çalışır” durumundaki prosesin geçebileceği durumlar ve nedenleri:

e) Proses kontrol bloğunda yer alan beş bilgi:

f) Proses ile iplik arasındaki iki fark:

g) Proses ile iplik arasındaki iki benzerlik:

h) İpliklerin kullanıcı düzeyinde gerçekleşmesinde görülebilecek iki problem:

i) Senkronizasyon:

j) Meşgul bekleme:

No	Ad - Soyad	İmza
----	------------	------

Soru 2: (15 puan) Proses_A, Proses_B ve paylaşılan bellek alanında yer alan bir “x” değişkeni bulunmaktadır. “**atomic_printf**” fonksiyonunun çalışması kesilemez olup, dış ortama çıkış üretmektedir.

```
int x = 0;
```

```
Proses_A()  
{  
    int i;  
    i = x;  
    i = i + 2;  
    x = i;  
    atomic_printf("x = %d\n", x);  
}
```

```
Proses_B()  
{  
    int i;  
    i = x;  
    i = i + 1;  
    x = i;  
    atomic_printf("x = %d\n", x);  
}
```

a) Proseslerin üreteceği çıkış değerleri ne olur (tüm olasılıkları yazın)?

b) Her iki proses tamamlandıktan sonra x’in alacağı son değer ne olur (tüm olasılıkları yazın)?

c) Bu proseslerle örneklenen duruma ne ad verilir? Açıklayın.

(Soru 3'te kullanılacak program)

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <pthread.h>
#include <sys/types.h>
#include <sys/wait.h>

#define COCUK_SAYISI 2
#define IPLIK_SAYISI 2

int sayac = 0;

void *altprogram(void *iplik_no) {
    sayac = sayac + 1;
    printf(" Altprogram: Proses kimlik no = %d Anne proses kimlik no = %d Iplik no = %d,
           Deger = %d \n", getpid(), getppid(), iplik_no, sayac);
    pthread_exit(NULL);
}

int main(void) {
    pthread_t iplikler[IPLIK_SAYISI];
    int cocuklar[COCUK_SAYISI];
    int rc, i, j;
    int f = 1;

    printf("Ana program: Proses kimlik no = %d Anne proses kimlik no: %d\n",
           getpid(), getppid());

    for (i = 0; i < COCUK_SAYISI; i++) {
        if (f > 0)
            f = fork();
        if (f == 0)
            break;
        else
            cocuklar[i] = f;
    }

    if (f == 0) {
        printf("Cocuk: Proses kimlik no: %d Anne kimlik no: %d sayac deger:%d\n",
               getpid(), getppid(), sayac);

        for(j = 0; j < IPLIK_SAYISI; j++)
            rc = pthread_create(&iplikler[j], NULL, altprogram, (void *)j);
    }

    else {
        for (i = 0; i < COCUK_SAYISI; i++)
            printf("Ana program: Cocuk[%d] kimlik no: %d yaratildi.\n", i, cocuklar[i]);
        wait(NULL);
    }

    return 0;
}
```

No	Ad - Soyad	İmza
----	------------	------

Soru 3. (25 puan)

(a) Önceki sayfada verilen programın çalışması esnasında oluşan proseslerin hiyerarşi ağacını çizin. İşletim sisteminin, çalıştırılan koda ilişkin ana prosese 1000 kimlik numarasını atadığı, kodun çalıştırıldığı kabuk (shell) programının kimlik numarasının 500 olduğu ve bu esnada başka hiçbir yeni prosesin çalışmadığı varsayımı altında, ağacın düğümlerine uygun proses kimlik numalarını yazın.

(b) Program çalıştırıldığında ekrana basılacak mesajları yazın. Bu mesajların “**ekrana basılma sıraları**” ve “**içerikleri**” hakkında yorum yapın.

c) **mx_end**: için yürütülmesi gereken kod parçası nedir?