

Ad Soyad	
Numara	
İmza	

1	2	3	4	5	6	7	8	9	Toplam
9	14	15	20	7	35				

Açıklamalar:

1. Toplam soru adedi dir.
2. Cevaplama süresi dakikadır.
3. Gerçeklemelerinizi verilen boşluklara yazınız.

1. (p) Ekran çıktısı

Aşağıdaki komut çıktılarını yazınız.

x=5, y=7 iken y += x + y > 10	y = 8
for (i = 0; i < 5; i++) printf("%c", 'a' + i);	Ekran: abcde
i = 99; while (i-- > 100);	i = 98

2. (p) Ekran çıktısı, Pointer

char s[] = "bar" iken bu as2 işlevi sonunda s'nin değeri nedir?

void as2(char *s) { char *p = s; while (*s) s++; for (s--; s>p; s--,p++) { char c = *s; *s = *p; *p = c; } }	Ekran: iki cevap da geçerlidir, - ab - rab
---	---

3. (p) Programlama, Dizgi işleme

Aşağıdaki çıktıyı üretecek **as3** programını yazınız.

Ekrana: This is a fall00 midterm question. his is a fall00 midterm question.T is is a fall00 midterm question.Th s is a fall00 midterm question.Thi is a fall00 midterm question.This is a fall00 midterm question.This s a fall00 midterm question.This i a fall00 midterm question.This is a fall00 midterm question.This is fall00 midterm question.This is a ... n. This is a fall00 midterm questio	Program: void as3 (char str[]) { /* test: as3("This is a fall00 midterm \ question."); */ int n, r, c; // strlen for (n = 0; str[n]; n++) ; for (r = 0; r < n; r++) { for (c = 0; c < n; c++) putchar(str[(r + c) % n]); putchar('\n'); } }
---	---

4. (p) Programlama, Modül

a) gerçel sayıyı en yakın tamsayıya çeviren **yuvarla** işlevini yazınız.

```
int yuvarla(float f) {  
    /* test:  
    int x = yuvarla(1.2), y = yuvarla(2.5), z = yuvarla(3.7);  
        // x=1, y=3 ve z = 4 olmalı    */  
    int i = f;  
  
    return (f - i) < 0.5 ? i : i+1;  
}
```

b) kendisine parametre olarak verilen **S**-boyutunu, **n**-adet mümkün mertebe eşit parçaya bölen ve böldüğü alt parçaların genişliklerini içeren dizi döndüren **linsz** programını yazınız.

```
int *linsz(int S, int n) {  
    /* test:  sz = linsz(10, 4);  
        // sz = [3 2 3 2] olmalı. sz = [2 2 2 4] olmamalı    */  
    int sz[N];  
  
    for (i = 0; i < n; i++) {  
        sz[i] = yuvarla((float) S / n);  
        S = S - sz[i];  
        n--;  
    }  
  
    return sz;  
}
```

5. (p) Düzeltin, Kodlama stili

Aşağıdaki kod parçalarındaki hataları düzeltin.

while (c = getchar() != EOF) putchar(c);	while ((c = getchar()) != EOF) putchar(c);
if (n > j && n < k)	if ((n > j) && (n < k))

6. (p) Programlama, Durum Makinesi

DNA alfabesinde dört sembol mevcuttur: **A**, **C**, **G**, **T**. Genom araştırmalarında bu sembollerin ardı ardına gelmeleri (sequence) özel anlamlar üretmektedir. *Inr* dna sequence'i olarak bilinen **YANXY** paterni bunlardan biridir. Burada **Y**, **C** veya **T**; **N** herhangi bir sembol ve **X**, **T** veya **A**'dır. Kendisine parametre olarak verilen dna sequence'inde *Inr* paterni arayan, bulursa indisini bulamazsa -1 döndüren **dna_inr** programını yazınız. Sembol **C** veya **T** ise **1**, diğer durumda **0** üreten **isY**; ve benzer biçimde sembol **T** veya **A** ise **1**, diğer durumda **0** üreten **isX** yardımcı işlevlerini kullanmak çözümünüzü kolaylaştıracaktır.

```
int isY(char ch) {
    return ((ch == 'C') |
            (ch == 'T')) ? 1 : 0;
}

int isX(char ch) {
    return ((ch == 'A') |
            (ch == 'T')) ? 1 : 0;
}

int dna_inr(char str[]) {
    /* test:
    dna_inr("CAATT"); // 0 uretir
    dna_inr("GATCAATAGTC"); // 6 uretir
    dna_inr("GATCAATAGGC"); // -1 uretir */
    int n, i, j, state;

    // strlen
    for (n = 0; str[n]; n++) ;

    for (j = 0; j < (n-5); j++) {
        state = 0;
        for (i = j; i < 5; i++)
            switch (state) {
                case 0: if (isY(str[i])) state = 1; break;
                case 1: if (str[i] == 'A') state = 2; break;
                case 2: state = 3; break;
                case 3: if (isX(str[i])) state = 4; break;
                case 4: if (isY(str[i])) return (i - state);
            }
    }

    return -1;
}

int dna_inr(char str[]) {
    /* test:
    dna_inr("CAATT"); // 0 uretir
    dna_inr("GATCAATAGTC"); // 6 uretir
    dna_inr("GATCAATAGGC"); // -1 uretir */
    int n, j;

    // strlen
    for (n = 0; str[n]; n++) ;
```

```
for (j = 0; j < (n-5); j++) {  
    if ( isY(str[i+0])    &&  
        str[i+1] == 'AA' &&  
        // i+2: herhangi bir karakter  
        isX(str[i+3])    &&  
        isY(str[i+4]) )  
        return j;          // bulduk  
  
    return -1;  
}
```