**سوال 1:** تعداد دستورات برنامه های زیر را تعیین کنید.

|  |
| --- |
| void **d(arrayElement \*x,** int **n)****{**int **i=1;**do **{** **x[i]+=2;** **i+=2;** **}**while(i<=n); **i=1;**while **(i<=(n/2))** **{** **x[i]+=x[i+1];** **i++;** **}****}**// end of d |

|  |
| --- |
| for(i=1;i<=n;i++)for(j=10;j<n-10)for(k=n;k>=0;k--)**x++;** |

|  |
| --- |
|  **i=1;**while **(i<=n)****{** **x++;**  **i++;** **}** |

**سوال 2:** پیچیدگی زمانی تابع بازگشتی زیر را بدست آورید:

**سوال 3:** اگر تخصیص حافظه به آرایه به صورت سطری باشد و دو آرایه x[1..9,1..2,1..4] و y[1..24] از یک نوع بوده و در حافظه روی هم تعریف شوند. خانه x(2,1,3) معادل کدام خانه Y می باشد؟

 آدرس شر وع <-- x(2,1,3) α+(2-1)\*(2-1+1)\*(4-1+1)+(1-1)\*(4-1+1)\*(3-1)=8+2=10

پس با آدرس شروع عضو Y[11] از یک مکان شروع می شوند.

**سوال 4:** در ریاضیات، به خصوص در نظریه ماتریس‌ها ، ماتریس نواری یک ماتریس خلوت است که درایه‌های خارج از یک نوار نسبتا باریک حول قطر اصلی صفر است .

به عبارت دیگر، ماتریس یک ماتریس نواری است اگر برای مقادیر ثابت  شرط زیر برقرار باشد :

مقادیر و  به ترتیب نیمه پهنای باند چپ ( پهنای نوار پایینی ) و راست ( پهنای نوار بالایی ) نامیده می شوند . یک ماتریس نواری با  یک ماتریس قطری است؛ یک ماتریس نواری با   یک ماتریس سه قطری( شکل 1)؛ یک ناتریس نواری با   یک ماتریس پنج وجهی و ... نامیده می‌شود .



اگر عنصر x[1,1] در خانه اول آرایه y ذخیره شده باشد، آدرس عناصر زیر را بدست آورید.( نوع عناصر دو آرایه یکسان است و x یک ماتریس سه قطری است).

X: array [1..60, 1..60] of word

1. X[10,10]
2. X[31,30]
3. X[40,42]

**سوال 5:** عبارت infix زیر را به postfix تبدیل کنید.

A\*(B+D)/E-F\*(G+H/k)

**سوال 6:** نتیجه عبارت postfix مقابل چیست؟

12 , 7, 3, -, /, 2,1,5,+,\*,+

**سوال 7:** فرض کنید دو صف Q1 و Q2 با مقادیر زیر داریم:

Q1: 20 , 15, 14, 31, 29, 16

Q2: 2, 10, 4, 1, 30, 7

اگر A و B ، عناصر صف باشند، محتوی صف Q3 بعد از دستورات زیر چیست؟



**سوال 8:** فرض کنید دو پشته در یک آرایه به اندازه (1…n) پیاده سازی شده اند. اگر Top یکی از پشته ها به خانه خالی و Top دیگری به خانه پر اشاره کند، شرط پر بودن آرایه چیست؟ (الگوریتمی برای بررسی پر بودن آرایه بنویسید)



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| n | n-1 | n-2 |  | 3 | 2 | 1 |

|  |  |
| --- | --- |
|  Stack2  Top2 |  Stack1 Top1 |