

---

5つすべての設問に答えよ。

## 1 構造体と疑似リスト

下記プログラムで、"12 26 39 51" と表示後、停止するには、(A),(B) にどのような式を書けば良いか答えよ。回答は、下線部に書き込むこと。

```
struct List {
    int data ;
    int next ;
} list[] = {
    { 12 , 2 } , { 39 , 3 } , { 26 , 1 } , { 51 , -1 }
} ; /*0*/          /*1*/          /*2*/          /*3*/
void main() {
    int i ;
    for( i = 0 ; _____ ; _____ ) {
        printf( "%d " , list[ i ].data ) ;
    }
}
```

## 2 処理速度のオーダー

処理速度の一般式が、 $T(N) = T_A + T_B 2^N + T_C N^2$  で表されるプログラムで、 $N = 20$  において、1秒であった。

1. 処理速度のオーダーを示せ。

2.  $N = 40$  になった場合の大まかな処理速度を予測せよ。

### 3 再帰呼び出し

```
int a[] = { 7 , 6 , 5 , 4 , 3 , 2 , 1 } ;
int sum( int i , int sz ) {
    if ( i >= sz ) {
        return 0 ;
    } else {
        return a[ i ]
            + sum( i*2 + 1 , sz )
            + sum( i*2 + 2 , sz ) ;
    }
}
void main() {
    printf( "%d\n" , sum( 0 , 7 ) ) ;
}
```

このプログラムの実行結果を示せ。

答えだけを記載するのではなく、処理の経過が分かるような情報 (書式は自由) を示すこと。

#### 4 メモリ管理

以下の様な、文字列を連結する関数 `str_append()` を作成し、動作確認を行う。

```
char* str_append( char* s1 , char* s2 )
{
    char* ans ;
    ans = (char*)malloc( _____ ) ;
    if ( _____ ) { _____(A)
        _____(B)
        strcpy( ans , s1 ) ;
        strcpy( ans + strlen( s1 ) , s2 ) ;
    }
    return ans ;
}
void main() {
    char* s3 = str_append( "Hello" , " , How are you!" ) ;
    _____(C)
}
```

下線部 (A) ~ (C) を埋め、その式を用いた理由を簡単に説明せよ。

(A) \_\_\_\_\_

(B) \_\_\_\_\_

(C) \_\_\_\_\_

## 5 処理速度

次のプログラムについて、以下の設問に答えよ。

```
int a[ 6 ] = { 1 , 4 , 4 , 2 , 2 , 2 } ;
int uniq( int ar[] , int sz )
{
    int i ;
    for( i = 0 ; i < sz ; i++ ) {
        int j ;
        for( j = i + 1 ; j < sz ; /*NONE*/ ) {
            if ( ar[ i ] == ar[ j ] ) {
                if ( j + 1 != sz )
                    ar[ j ] = ar[ sz - 1 ] ;
                sz-- ;
            } else {
                j++ ;
            }
        }
    }
    return sz ;
}
void main()
{
    int s = uniq( a , 6 ) ;
    int i ;
    for( i = 0 ; i < s ; i++ )
        printf( "%d " , a[ i ] ) ;
}
```

1. このプログラムで表示される内容答えよ。
2. もし、配列a[] の中に同じ値のデータが無く、データが  $N$  件あった時の、関数 `uniq( a , N )` の処理時間のオーダーを求めよ。  
ただし、導出過程の説明をすること。