

## 1 基本理解 (必須問題)

以下のようなプログラムを作成した。設問に答えよ。

```
struct List {
    int      data ;
    struct List* next ;
} ;
struct List* cons( int x , struct List* n ) {
    struct List* ans ;
    ans = (struct List*)malloc( sizeof( struct List ) ) ;
    if ( ans != NULL ) {
        ans->data = x ;
        ans->next = n ;
    }
    return ans ;
}
struct List* decompose( int x )
{
    struct List* ans = NULL ;
    int      i ;

    while( x > 1 ) {
        for( i = 2 ; i <= x ; i++ ) {
            if ( x % i == 0 ) {
                ans = cons( i , ans ) ;
                x = x / i ;
                break ;
            }
        }
    }
    return ans ;
}
void print( struct List* p ) {
    for( ; p != NULL ; p = p->next )
        printf( "%d " , p->data ) ;
}
void main() {
    print( decompose( 6 ) ) ;
}
```

1. `decompose(6)` が呼び出された時に、下線 A ~ H までは、どのような順序で実行されたか、記号によって順序を示せ。

2. `decompose(6)` の結果のリスト構造をイメージ図によって示せ。

3. このプログラムによって表示される内容を示せ。

4. プログラム中の下線部  $\alpha \sim \delta$  の型を述べよ。

$\alpha$  \_\_\_\_\_

$\beta$  \_\_\_\_\_

$\gamma$  \_\_\_\_\_

$\delta$  \_\_\_\_\_

## 2 説明問題 (選択)

一般的にリスト構造を採用した場合、どのような利点や欠点があるか？

最低限 5つの項目をあげて、利点欠点を述べよ。

ただし『××に比べてリスト構造は、○○である点が利点である』といった様に、客観的な表現を用いること。

1.

2.

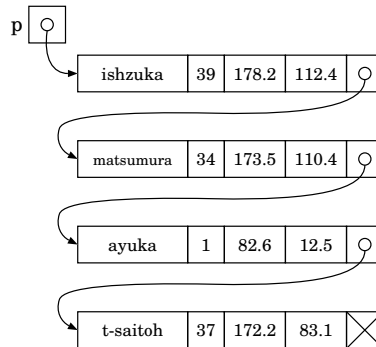
3.

4.

5.

### 3 データ構造の設計 (選択)

名前・年齢・身長・体重からなるデータベースのプログラムをリスト構造により記述したい。



1. 上記イメージ図のようなデータを格納できるリスト構造の宣言文を記述せよ。

2. リスト構造の先頭のポインタを引数とし、次の2つの条件のどちらかを満たすデータについて、名前を表示したい。

- 年齢が10歳未満、
- 肥満度 =  $\frac{\text{体重} - (\text{身長} - 100)}{\text{体重}} \times 100[\%]$  が20%以上

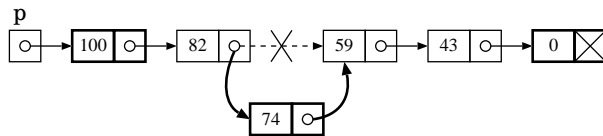
関数名 `young_or_fat()` にて、関数を作成せよ。

#### 4 挿入のある処理 (選択)

入力されたデータを、常に降順になるように、データを挿入する関数 `insert(struct List*,int)` を作成せよ。

処理は、リストの先頭から、与えられた数より小さいデータが見つかるまで移動し、その中間に新しいリストを挿入する。

入力されるデータは、1から99までの数とし、処理を簡単にするために、最初に100と0のデータを入れておく。データ構造は、設問1と同じとする。



```
void insert( struct List* p , int x ) {  
    /* この部分を記述せよ */  
}  
void main() {  
    int x ;  
    struct List* ls = cons( 100 , cons( 0 ) ) ;  
    while( scanf( "%d" , &x ) == 1 ) {  
        insert( ls , x ) ;  
    }  
}
```

## 5 応用記述問題 (選択)

1つの文字列の中に、':'を区切り文字として、複数の文字列が書き並べられている。この文字列中の':'を'\0'に書き換え、区切られた文字列の先頭アドレスを要素とするリストを生成したい。

文字列を引数とし、生成された  
StrList のリスト構造を返す関数  
struct StrList\* split( char\* str )  
を作成せよ。

```
/* 文字ポインタのリスト */  
struct StrList {  
    char* str ;  
    struct StrList* next ;  
};
```

