

以下の6つの設問の中から、5つを選んで解答せよ。

1 基本問題 (20)

```

int stack[ 10 ] ;
int sp = 0 ;

int var[ 3 ] = {
    11 , 22 , 33 } ;

int eval( char s[] ) {
    for( int i = 0 ; s[ i ] != '\0' ; i++ ) {
        if ( '0' <= s[ i ] && s[ i ] <= '9' ) {
            push( s[ i ] - '0' ) ;
        } else if ( 'A' <= s[ i ] && s[ i ] <= 'C' ) {
            push( var[ s[ i ] - 'A' ] ) ;
        } else {
            int r = pop() ;
            int l = pop() ;
            switch( s[ i ] ) {
                case '+' : push( l + r ) ; break ;
                case '-' : push( l - r ) ; break ;
            }
        }
    }
    return pop() ;
}

void push( int x ) {
    stack[ sp++ ] = x ;
}

int pop() {
    return stack[ --sp ] ;
}

int main() {
    printf( "%d\n" , eval("31+") ) ; /*A*/
    printf( "%d\n" , eval("C21+-") ) ; /*B*/
    return 0 ;
}
    
```

設問1 実行結果を答えよ。

(A) _____ , (B) _____ (5x2)

設問2 eval() に与えるような式の書き方を何と云うか答えよ。

(C) _____ (5)

設問3 一般的な式の書き方 $3*(2+1)$ を設問2の書き方で示せ。

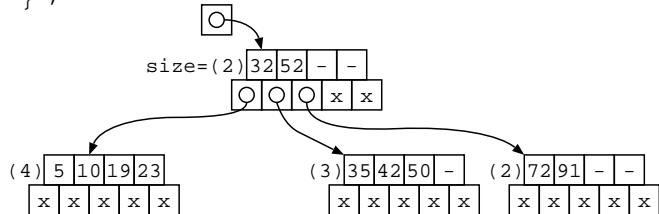
(D) _____ (5)

2 説明問題 (20)

- AVL木における枝の回転処理について具体的なイメージ図を交えて説明せよ。(10)
- 図に示すB木に、一般的な追加処理で100と30を加えた後のイメージ図を示せ。(10)

```

#define SIZE 2
struct BTree {
    int size ;
    int data[ SIZE * 2 ] ;
    struct BTree* ptr[ SIZE * 2 + 1 ] ;
} ;
    
```



3 両端 Queue(20)

```

struct Deque {
    struct Deque* prev ;
    int data ;
    struct Deque* next ;
} ;

int shift( struct Deque* q ) {
    struct Deque* d = q->next ;
    ~~~~~(a)

    int ans = d->data ;
    q->next = d->next ;
    d->next->prev = d->prev ;
    free( d ) ;
    return ans ;
}

int main() {
    struct Deque* top = dq_cons( NULL , -1 , dq_cons( NULL , 11 , NULL ) ) ;
    top->prev = top->next ;
    top->next->prev = top ;
    top->next->next = top ;
    /* (A) */
    unshift( top , 22 ) ;
    /* (B) */
    printf( "%d\n" , shift( top ) ) ;
    /* (C) */
    return 0 ;
}

struct Deque* dq_cons( struct Deque* p , int x ,
                      struct Deque* n ) {
    struct Deque* ans ;
    ans = (struct Deque*)malloc(sizeof(struct Deque)) ;
    if ( ans != NULL ) {
        ~~~~~(b)
        ~~~~~(c)
        ans->prev = p ; ans->data = x ; ans->next = n ;
        ~~~(d)
    }
    return ans ;
}

void unshift( struct Deque* q , int x ) {
    struct Deque* t = dq_cons( q , x , q->next ) ;
    t->next->prev = t ;
    q->next = t ;
}
    
```

設問 1 (a) ~ (d) の型を答えよ。
 (a) _____ (b) _____
 (c) _____ (d) _____ (2x4)
 設問 2 (C) の実行結果を答えよ。
 _____ (2)
 設問 3 (A), (B) の時点での
 top の先のデータ構造のイメージ図を
 描け。(5x2)

4 両端 Queue の記述問題 (25)

前設問の両端 Queue の末尾への追加削除を行う関数 push(), pop() の下線部にふさわしい処理を答えよ。

```

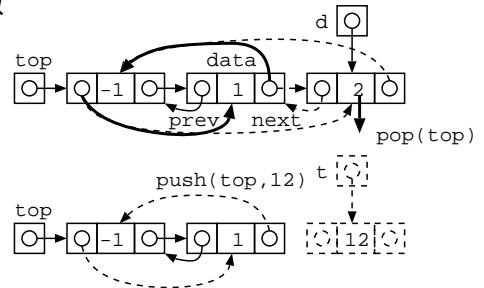
int pop( struct Deque* q ) {
    // 末尾を取り出す
    struct Deque* d = _____ ; (A)4

    int ans = d->data ;

    q->prev = _____ ; (B)4
    d->prev->next = _____ ;
    free( d ) ; _____ (C)4
    return ans ;
}

void push( struct Deque* q , int x ) {
    // 末尾に入れる
    struct Deque* t = _____ ; (D)4

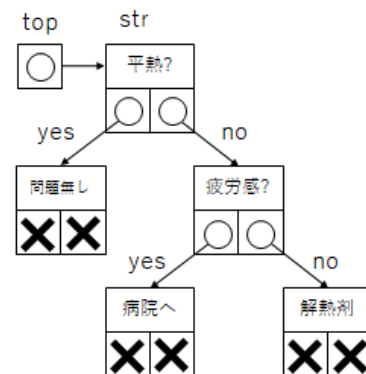
    _____ ; (E)4
    q->prev = t ;
}
    
```



5 意思決定木 (20)

図に示すような意思決定木の処理を行うプログラムを作成する。

- (1) この処理のためのデータ構造の定義を答えよ。(2) また質問文の木を引数とし、質問を繰返しなが、最終判断結果を表示するための関数 `question()` を作成せよ。(7+13)
yes/no の入力は、`scanf("%d",...)` で 1=yes,0=no で判断すること。



6 コンパイラと構文解析 (20)

1. 以下の BNF 規則において、設問 [1] ~ [3] は<パラメタ指定>として適切か ○, × で答えよ。

<パラメタ指定> ::= <パラメタ> | (<パラメタ指定> , <パラメタ>) ;
<パラメタ> ::= <英字> | <パラメタ> <英字> ;
<英字> ::= a | b | c | d | e | f | g | h | i ;

[1] (abc) ____ [2] ((abc,def),ghi) ____ [3] (abc,(def)) ____ (3x3)

2. 一般的な C 言語のソースプログラムから機械語が生成されるまでの内部処理について説明せよ。(11)