

3. Some languages (like C++) allow the programmer can define local variables anywhere in a block and that variable name is scoped to that block from that point on to the end of the block. Needless to say, this complicates the compiler's scoping and type checking mechanisms.

Consider the following valid C++ program fragment:

```
int main( char *argv[], int argc )
{
    int i = 2;

    while ( i == 2 )
    {
        i--;
        int i = 2;
        i++;
        if ( i > 2 )
        {
            i--;
            int i = 2;
            i++;
            cout << i << " ";           // Output the current value of i followed by a space
        }
        cout << i << " ";           // Output the current value of i followed by a space
    }
    cout << i << endl;           // Output the current value of i followed by a newline

    return 0;
}
```

What gets printed? _____

4. Indicate whether the following expressions are

- A. legal (no compiler error) or
- B. illegal (compiler error).

```
int a[10];
int *iPtr;
```

```
&a[4] = iPtr;      _____
```

```
iPtr = &a[4];      _____
```

```
*(iPtr - 3) = a[3]; _____
```

```
*&a[3] = *iPtr;   _____
```

```
&a = iPtr + 420;  _____
```

```
*(a+2) = 2[iPtr]; _____
```

What question would you most like to see on the Midterm?