

Announcements

Tenure track search continues
confidentiality important
can talk to others on campus
do NOT say anything to anyone off campus
Office hours next week subject to adjustment
some telephoning of candidates
other candidates should not know
they are not on the initial interview list!!!

Projects, Supplemental Problems: the following are required
For each function, need Comments, Pre/post conditions
Format reflects structure
Testing (both black box and white box testing)
Program under development without comments, formatting
==> no help possible (don't ask)
Submission without one of these ==> no grade

Definitions in MyroC

```
typedef struct
{
    unsigned char R;
    unsigned char G;
    unsigned char B;
} Pixel
```

```
typedef struct
{
    int height; //<= 266
    int width;  //<= 427
    Pixel pix_array[266][427];
} Picture;
```

Consider statements about
height and width:

1. setting height=266,
width=427 is always safe
2. any values for height, width
will work

- A. 1 and 2
- B. 1, not 2
- C. 2, not 1
- D. neither 1
nor 2

A white Pixel may be declared as

```
Pixel whitePix = {255, 255, 255}; // white pixel
```

Which, if any, of the following statements define a picture of the prescribed color?

A. `Picture pic1 = {200, 300, {{whitePix}}};` //white picture

B. `Picture pic2 = {200, 300, {{{255,255,255}}}};` //white picture

1. A and B 2. A, not B 3. B, not A 4. neither
A nor B

A black Pixel may be declared as

```
Pixel blackPix = {0, 0, 0}; // black pixel
```

Which, if any, of the following statements define a picture of the prescribed color?

- A. `Picture pic3 = {200, 300, {{{0}}}};` `//black picture`
B. `Picture pic4 = {200, 300, {{blackPix}}};` `//black picture`

1. A and B 2. A, not B 3. B, not A 4. neither
A nor B

Recall MyroC defines a Pixel:

```
typedef struct
{
    unsigned char R;
    unsigned char G;
    unsigned char B;
} Pixel
```

Suppose variable `pix` is declared as a `Pix``el`, and suppose one wants to increase its red value by 10, except that it may not exceed 255 as the maximum red value. Which of the following approaches work reliably:

approach 1

```
pix.R += 10;
if (pix.R > 255)
    pix.R = 255;
```

approach 2

```
int value = pix.R+10;
if (value > 255)
    value = 255;
pix.R = value;
```

A. both 1
and 2

B. 1,
but not 2

C. 2,
but not 1

D. neither 1
nor 2

According to the lab (with a citation), converting a pixel to gray scale involves setting RGB values of a pixel to 30% of the red value + 59% of the green value + 11% of the blue value.

Which of the following code segments accomplish this?

approach 1

```
pix.R = 0.3*pix.R + 0.59*pix.G + 0.11*pix.B;  
pix.G = 0.3*pix.R + 0.59*pix.G + 0.11*pix.B;  
pix.B = 0.3*pix.R + 0.59*pix.G + 0.11*pix.B;
```

approach 2

```
int value = 0.3*pix.R + 0.59*pix.G + 0.11*pix.B;  
pix.R = value;  
pix.G = value;  
pix.B = value;
```

- | | | | |
|--------------------|--------------------|--------------------|-----------------------|
| A. both 1
and 2 | B. 1,
but not 2 | C. 2,
but not 1 | D. neither 1
nor 2 |
|--------------------|--------------------|--------------------|-----------------------|