629645
Suppose you are writing a Java program to be used to manage information about the inventory in a store. The store’s products include pagers, cell phones, and cameras. The cell phones are used for both communication and taking pictures, the pagers are only used for communication, and the cameras are used for taking pictures.

Assume that a Product class has been defined. Which of the following is the best way to represent the remaining data?

a. Define two subclasses of the Product class: Communicator and PictureTaker. Define two subclasses of the Communicator class: Pager and CellPhone; and define two subclasses of the PictureTaker class: CellPhone and Camera.

*b. Define three subclasses of the Product class: Pager, CellPhone and Camera. Also define two interfaces: Communicator and PictureTaker. Define the Pager and CellPhone classes to implement the Communicator interface, and define the CellPhone and Camera classes to implement the PictureTaker interface.

c. Define five new classes, not related to the Product class: Pager, CellPhone, Camera, Communicator, and PictureTaker.

d. Define five subclasses of the Product class: Pager, CellPhone, Camera, Communicator, and PictureTaker.

e. Define two subclasses of the Product class: Communicator and PictureTaker. Also define three interfaces: Pager, CellPhone, and Camera. Define the Communicator class to implement the Pager and CellPhone interfaces, and define the PictureTaker class to implement the CellPhone and Camera interfaces.

f. "
g. "
h. "
i. "
j. "

General Feedback:

Answer A is the easiest to rule out -- it gives `CellPhone` two superclasses, which is not permitted in Java.

C is not the best choice because it doesn't capture the relationships among the various items. D captures some of the relationships but not all (for example, it's missing the relationship between `PictureTaker` and `Camera`).

Answers B and E are very similar -- they focus on the question of which items should be modeled by interfaces and which by classes. B is a better choice for several reasons. Pagers, phones, and cameras all have both behaviors and data -- for example, the list of phone messages for a phone and the ability to take pictures for both phones and cameras -- while PictureTaker and Communicator correspond to behaviors. In addition, defining PictureTaker as an interface allows us to capture the fact that a cell phone is a product, a picture taker, *and* a communicator.

632075
Consider the following short program, which does not meet all institutional coding standards:

```
void vCodeString(char szText[ ]); /* First line */
#include <stdio.h>
#include <string.h>
#define MAX_LEN 12
int main(void)
{
     char szData[MAX_LEN];
     printf("Enter some text to code: ");
     scanf("%s", szData);
     vCodeString(szData); /* Line 8 */
     printf("Coded string is %s\n", szData);
}
void vCodeString(char szText[ ])
{
     int i = -1;
     while(szText[++i])
     {
          szText[i] += (char)2;
     }
}
```

I would like to combine lines 8 and 9 into

`printf("Coded string is %s\n", vCodeString(szData));`

Unfortunately, this gives me a syntax error. Why?

a. The %s should be replaced by %c.

b. The parameter to vCodeString should be szText

c. The parameter to vCodeString should be shown as szData[ ].

d. printf can only match %s to a variable, not to a function.

*e. vCodeString does not return a value.

f. "
g. "
h. "
i. "
j. "

General Feedback:

The printf function cannot take as a parameter a function which does not return a value

635015
The binary value 11011101 when coverted into a decimal value from two's complement notation is:

*a. -35

b. 35

c. 221

d. -221

e. -33

f. "
g. "
h. "
i. "
j. "

General Feedback:

Conversion of negative numbers in two's complement into decimal requires a bit flip, followed by the binary addition of 1. This value is then interpreted as an unsigned binary value.