

PES, Section 2.2
C Data Types

1. How many bits are in an unsigned long?

32

2. What is the range of values that an unsigned short can have?

An unsigned short is 16-bits long. Therefore the range of values is $[0, 2^{16} - 1]$

3. How do you represent a 1-bit value in C?

**Use a char; 7 bits are unavoidably wasted.
You could also use an enumerated data type.**

4. Suppose that you have an unsigned short that will represent the age of a person. What is an appropriate name for the variable?

**usage;
(or us_age).
(pre-pend “us” so that you always know that the data type is unsigned short)**

5. Does use of the signed or unsigned ‘int’ data type affect portability? Explain.

Yes. C does not define the number of bits in the ‘int’ data type, so it varies from compiler-to-compiler and platform to platform.

6. Suppose that you have a variable foo that will represent a value in the range -1 to 255. What data type should you use and why?

signed short;
- **“signed” is necessary to represent negative values**
- **9 bits are required to represent 257 discrete values. The smallest C data type that can represent 9-bit values is “short” (16 bits).**