



Problem of the Week

Problem A and Solution

Archaeological Ages

Problem

You are an archaeologist, and have made a startling series of discoveries right around the corner from your office.

You have taken the pieces back to the lab, and are trying to figure out how old your objects are. Your rival, Indianapolis Jane, has tampered with your artifact dating machine. The ages it is computing are correct but are hard to compare. Here are the results from the machine:

Artifact #1 is 50 decades, 100 weeks, and 48 hours old

Artifact #2 is 3 centuries, 4 decades, and 6 years old

Artifact #3 is 2 centuries, 28 decades, and 52 weeks old

You want to present these artifacts to your colleagues in order from oldest to newest. In which order should you present them? Justify your answer.

NOTE: A *decade* is equal to 10 years, and a *century* is equal to 100 years.

Solution

One way to solve this problem is to estimate the age of each artifact. To compare the ages, we need a common unit. We know that 1 decade is equal to 10 years, and that 1 century is equal to 100 years. We know that each year has approximately 52 weeks and each day has 24 hours. So we can approximate the age of each artifact in years.

- Approximate age of **Artifact #1**:

We see that 50 decades is equal to $50 \times 10 = 500$ years, and 100 weeks is approximately 2 years, and 48 hours is equal to 2 days which is insignificant in the calculation of the number of years in the age. We can estimate the age of **Artifact #1** at approximately $500 + 2 = 502$ years old.

- Approximate age of **Artifact #2**:

Since 3 centuries is equal to $3 \times 100 = 300$ years, and 4 decades is equal to $4 \times 10 = 40$ years, we can estimate the age of **Artifact #2** at approximately $300 + 40 + 6 = 346$ years old.

- Approximate age of **Artifact #3**:

Since 2 centuries is equal to $2 \times 100 = 200$ years, and 28 decades is equal to $28 \times 10 = 280$ years, and 52 weeks is approximately 1 year, we can estimate the age of **Artifact #3** at approximately $200 + 280 + 1 = 481$ years old.

So the oldest item is **Artifact #1**, the next oldest is **Artifact #3**, and the newest is **Artifact #2**.



Teacher's Notes

Finding a common unit of measurement can be important in our every day lives. We often find situations where we need to compare or equate the sizes of items that have been measured using different units. For example, we might have a recipe that uses metric units such as millilitres and Celsius but our own kitchens contain measuring cups and ovens that have Fahrenheit settings or vice versa.

In many cases it is possible to convert from one unit to another precisely. For example, with metric units it is very easy to convert among the base unit and units with prefixes such as *milli*, *centi* and *kilo*. We can also precisely convert between units that have been defined in relation to each other, such as $1 \text{ minute} = 60 \text{ seconds}$.

In other cases we often estimate the relationship between units of measurement, especially when we do not need to be precise. For example, we estimate that $1 \text{ year} = 52 \text{ weeks}$, but this is not an accurate relationship, since $1 \text{ week} = 7 \text{ days}$ and $7 \times 52 = 364$. Normally there are 365 days in a year, so a more precise estimate would be to say that $1 \text{ year} = 52\frac{1}{7} \text{ weeks}$. However, this is not accurate either since leap years have 366 days.

Estimation is perfectly acceptable when more precise units of measurement are either irrelevant or meaningless. In this problem, it is not necessary to know the exact age of any of the artifacts to be able to determine which one is the oldest. We use terms like *circa* when referring to dates to give historical context without necessarily knowing an exact year. We measure ages in weeks or years rather than minutes since knowing someone is 525 600 minutes old is usually not that informative. There are times when precise measurement is important, but there are occasions where approximation is valid and useful.