# An FPTAS for the Subset Sum Problem



## Approximation Scheme, PTAS, FPTAS

- An **approximation scheme** for a problem is a $(1+\epsilon)$-approximation algorithm, for any $\epsilon > 0$.
- An approximation scheme is a **polynomial-time approximation scheme (PTAS)** if, for any fixed $\epsilon > 0$, the scheme runs in polynomial time in the size $n$ of its input instance.

  **Example:** $O\left( n^{\frac{2}{\epsilon}} \right)$
- An approximation scheme is a **fully polynomial-time approximation scheme (FPTAS)** if, for every $\epsilon > 0$, the scheme runs in polynomial time both in $\frac{1}{\epsilon}$ and in the size $n$ of its input instance.

  **Example:** $O\left( \epsilon^{-2} n^{4} \right)$

# Subset Sum Problems

Let $S = \{x_1, x_2, \ldots, x_n\}$ be a set of positive integers. The **sum of** $S$ is:

$$\Sigma S = \sum_{i=1}^{n} x_i \qquad \text{(where } \Sigma\emptyset = 0\text{)}.$$

## Decision Problem

Given a set $S$ of positive integers and a positive integer $T$, is there a subset $A \subseteq S$ such that $\Sigma A = T$?

**Instance:** $S = \{1, 5, 6, 12, 24, 31\}$ and $T = 30$.

## Optimization Problem

Given a set $S$ of positive integers and a positive integer $T$, find a subset $A \subseteq S$ such that $\Sigma A$ is as large as possible and $\Sigma A \leq T$.



# Notation

- If $L$ is a list of integers and $x$ is an integer, $L \bigoplus x$ denotes the list of integers derived from $L$ by increasing each element of $L$ by $x$.

  **Example:** if $L = < 1, 2, 3, 5, 9 >$, then $L \bigoplus 2 = < 3, 4, 5, 7, 11 >$.

- If $S$ is a set of integers and $x$ is an integer, the definition of $S \bigoplus x$ is similar: $S \bigoplus x = \{s + x \mid s \in S\}$.

- mergeLists($L_1$, $L_2$) returns the sorted list that is the merge of its two sorted input lists $L_1$ and $L_2$ with duplicate values removed.

- removeGreater($L$, $T$) returns the sorted list with the elements of the sorted input list $L$ that do not exceed $T$.

# Exact Algorithm

```
int exactSubsetSum( Set<int> S, int T )
{   // Let S = {x_1, x_2, ..., x_n}.
    OrderedList<int> L = < 0 >;
    for ( int i = 1;   i <= S.size();   i++ )
    {
        L = mergeLists(L, L ⊕ x_i);
        L = removeGreater(L, T);
    }
    return L.maxElement();
}
```

**Running time:** $O(|S| \times \max(|L|))$



# Example: $S = \{10, 14, 15, 19\}$, $T = 35$

$L = < 0 >$

$L \oplus 10 = < 10 >$

$L = < 0, 10 >$

$L \oplus 14 = < 14, 24 >$

$L = < 0, 10, 14, 24 >$

$L \oplus 15 = < 15, 25, 29, 39 >$

$L = < 0, 10, 14, 15, 24, 25, 29, 39 >$

$L = < 0, 10, 14, 15, 24, 25, 29 >$

$L \oplus 19 = < 19, 29, 33, 34, 43, 44, 48 >$

$L = < 0, 10, 14, 15, 19, 24, 25, 29, 33, 34, 43, 44, 48 >$

$L = < 0, 10, 14, 15, 19, 24, 25, 29, 33, 34 >$ **Result:** 34

# Proposition

exactSubsetSum is an exponential algorithm that solves the Subset Sum problem.

# Proof

Let $S = \{x_1, x_2, \ldots, x_n\}$ and $T \geq 1$.

After step $i$ (for every $i = 1, \ldots, n$):

- $L$ is a sorted list containing the sums of all subsets of

$$\{x_1, x_2, \ldots, x_i\}$$

  that do not exceed $T$;
- $L$ can have $2^i$ elements.



# List Trimming

Let $L$ be a list of positive integers and $\delta \in ]0, 1[$.

Element $y$ is removed from $L$ if there is an element $z$ in $L$ such that:

$$\boxed{z \leq y \quad \text{and} \quad \frac{y}{1+\delta} \leq z}$$

If $L = < 10, 11, 12, 15 >$ and $\delta = 0.1$, trim$(L,\ 0.1) = < 10, 12, 15 >$.

$$\boxed{10}$$

$$11 \qquad \frac{11}{1.1} = 10 \leq 10$$

$$\boxed{12} \qquad \frac{12}{1.1} > 10.9 > 10$$

$$\boxed{15} \qquad \frac{15}{1.1} > 13.6 > 12$$

# List Trimming

```
OrderedList<int> trim( OrderedList<int> L, real δ )
{   // Let L = < y_1, y_2, ..., y_k >, for some k > 0.
    OrderedList<int> result = < y_1 >;
    int last = y_1;
    for ( int i = 2;   i <= L.size();   i++ )
        if ( y_i > last * (1 + δ) )
        {
            result.add(y_i);
            last = y_i;
        }
    return result;
}
```

**Running time:** $O(|L|)$



# Approximation Scheme for $\epsilon \in ]0, 2]$

```
int approxSubsetSum( Set<int> S, int T, real ε )
{   // Let S = {x_1, x_2, ..., x_n}.
    int n = S.size();
    OrderedList<int> L = < 0 >;
    for ( int i = 1;   i <= n;   i++ )
    {
        L = mergeLists(L, L ⊕ x_i);
        L = trim(L, ε/(2 * n));
        L = removeGreater(L, T);
    }
    return L.maxElement();
}
```

**Running time:** $O(|S| \times \max(|L|))$

## Example: $S = \{10, 14, 15, 19\}$, $T = 35$, $\epsilon = 0.3$

$L = < 0 >$ $\delta = \frac{0.3}{2 \times 4} = 0.0375$

$L \oplus 10 = < 10 >$

$L = < 0, 10 >$

$L \oplus 14 = < 14, 24 >$

$L = < 0, 10, 14, 24 >$

$L \oplus 15 = < 15, 25, 29, 39 >$

$L = < 0, 10, 14, 15, 24, 25, 29, 39 >$

$L = < 0, 10, 14, 24, 29 >$

$L \oplus 19 = < 19, 29, 33, 43, 48 >$

$L = < 0, 10, 14, 19, 24, 29, 33, 43, 48 >$

$L = < 0, 10, 14, 19, 24, 29, 33 >$ **Result:** 33



## approxSubsetSum is an Approx. Scheme

Let $S = \{x_1, x_2, \ldots, x_n\}$ and $T \geq 1$.

After step $i$ (for every $i = 1, \ldots, n$), let $Y_i$ be the set containing the sums of all subsets of

$$\{x_1, x_2, \ldots, x_i\}$$

that do not exceed $T$.

Then, for every $y \in Y_i$, there is an element $z \in L_i$ such that:

$$z \leq y \leq \left( 1 + \frac{\epsilon}{2n} \right)^i z.$$

Therefore, if $y^*$ is the optimal solution, there is $z^* \in L_n$ such that:

$$z^* \leq y^* \leq \left( 1 + \frac{\epsilon}{2n} \right)^n z^*.$$

# Proof

**Step 1**

- Since $L_0 = < 0 >$, "$Y_1 \subseteq L_1$". So, let $z = y$, and $z \leq y < z\,(1+\delta)$.

**Step $i+1$ (for any $i \geq 1$)**

- If $y \in L_{i+1}$, let $z = y$. So, $z \leq y < z\,(1+\delta)^{i+1}$.
- If $y \notin L_{i+1}$,
  - if $y \in L_i$, by constr., $(\exists z \in L_{i+1})\; z < y \leq z\,(1+\delta) < z\,(1+\delta)^{i+1}$.
  - if $y \notin L_i$, by induction hypothesis, $(\exists z \in L_i)\; z \leq y \leq z\,(1+\delta)^i$.
    * When $z \in L_{i+1}$, $z \leq y < z\,(1+\delta)^{i+1}$.
    * When $z \notin L_{i+1}$, by constr., $(\exists w \in L_{i+1})\; w < z \leq w\,(1+\delta)$.
      Therefore, $w < y$ because $w < z \leq y$, and
      $y \leq z\,(1+\delta)^i \leq w\,(1+\delta)\,(1+\delta)^i$ implies $y \leq w\,(1+\delta)^{i+1}$.



# approxSubsetSum is an Approx. Scheme

$$y^* \leq \left(1 + \frac{\epsilon}{2n}\right)^n z^*$$

$$\begin{aligned}
\left(1 + \frac{\epsilon}{2n}\right)^n &\leq e^{\frac{\epsilon}{2}} && \text{because } \frac{\partial}{\partial n}\left(1 + \frac{x}{n}\right)^n > 0 \\
&&& \text{and } \lim_{n \to +\infty}\left(1 + \frac{x}{n}\right)^n = e^x \\
&\leq 1 + \frac{\epsilon}{2} + \left(\frac{\epsilon}{2}\right)^2 && \text{because } e^x \leq 1 + x + x^2 \text{ when } |x| \leq 1 \\
&&& \text{and } \epsilon \in\, ]0, 2] \\
&\leq 1 + \epsilon && \text{because } \epsilon \in\, ]0, 2]
\end{aligned}$$

$$\boxed{y^* \leq (1+\epsilon)\, z^*}$$

# approxSubsetSum is an FPTAS (1)

At the end of each step, $L = < y_1, y_2, \ldots, y_k >$ and $\frac{y_{j+1}}{y_j} > 1 + \frac{\epsilon}{2n}$.

Therefore,

$$
\begin{aligned}
|L| &\leq 2 + \log_{1+\frac{\epsilon}{2n}} T \\
&= 2 + \frac{\ln T}{\ln\left(1 + \frac{\epsilon}{2n}\right)} \\
&\leq 2 + \frac{\left(1 + \frac{\epsilon}{2n}\right) 2n \ln T}{\epsilon} \\
&\leq 2 + \frac{4n \ln T}{\epsilon}
\end{aligned}
$$

$$\log_b a = \frac{\log_c a}{\log_c b}$$

$$\ln(1+x) \geq \frac{x}{1+x} \quad \text{when} \quad x > -1$$

$$\epsilon \in \,]0, 2] \;\Rightarrow\; \frac{\epsilon}{2n} \leq 1$$



# approxSubsetSum is an FPTAS (2)

At the end of each step,

$$|L| \leq 2 + \frac{4n \ln T}{\epsilon}.$$

Hence,

$$\text{approxSubsetSum}(S, T, \epsilon) = O\left(\frac{|S|^2 \ln T}{\epsilon}\right).$$