

Постановка задачи:

Структура вычислительной системы (ВС) задана в виде набора модулей, соединённых последовательно (рис. 1). Каждый модуль состоит из аппаратного компонента и программного компонента. Известны наборы вариантов устройств и программ, которые могут присутствовать в модуле в качестве соответственно аппаратного и программного компонентов. Для каждого варианта устройства или программы известны надёжность и стоимость.

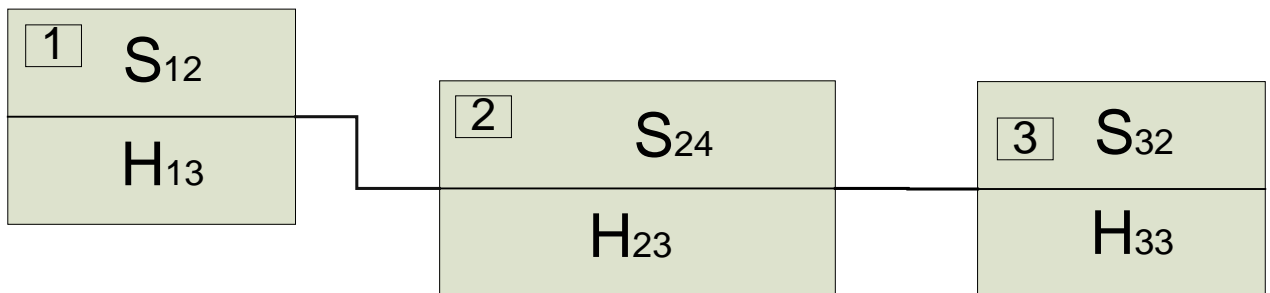


Рисунок 1. Структура вычислительной системы.

Необходимо для каждого модуля системы выбрать такой вариант аппаратного и программного компонента, при которых надёжность системы максимальна при ограничении на стоимость. Под надёжностью системы будем понимать вероятность ее безотказной работы в течение некоторого заданного промежутка времени.

Для формального описания структуры ВС введём следующие обозначения:

- n – количество модулей ВС;
- U_i – i -ый модуль системы;
- p_i, q_i – количество доступных версий соответственно аппаратного и программного компонентов в модуле U_i ;
- H_{ij} – j -ая версия аппаратного компонента модуля $U_i, j \in [1, p_i]$;
- S_{ij} – j -ая версия программного компонента модуля $U_i, j \in [1, q_i]$;
- R_{ij}^{hw}, C_{ij}^{hw} – надёжность и стоимость j -ой версии аппаратного компонента модуля U_i ;
- R_{ij}^{sw}, C_{ij}^{sw} – надёжность и стоимость j -ой версии программного компонента модуля U_i ;

- x_{ij} – количество экземпляров H_{ij} в модуле U_i ; $x_{ij} = \{0;1\}$. $\forall i \sum_{j=1}^{p_i} x_{ij} = 1$;
- y_{ij} – количество экземпляров S_{ij} в модуле U_i ; $y_{ij} = \{0;1\}$. $\forall i \sum_{j=1}^{q_i} y_{ij} = 1$;
- C_{System}^{max} – максимальная допустимая стоимость конфигурации РВС РВ;
- $Systems$ – множество всевозможных конфигураций РВС РВ.

Конфигурация i -ого модуля однозначно определяется парой $U_i = \{H_{ij}, S_{ij}\}$. Система $System = \{U_1, \dots, U_n\}$.

Стоимость ВС можно вычислить по формуле:

$$C_{System} = \sum_{i=1}^n C_i = \sum_{i=1}^n \left(\sum_{j=1}^{p_i} x_{ij} \cdot C_{ij}^{hw} + \sum_{j=1}^{q_i} y_{ij} \cdot C_{ij}^{sw} \right).$$

Надёжность ВС равна:

$$R_{System} = \prod_{i=1}^n R_i = \prod_{i=1}^n \left(\left(\sum_{j=1}^{p_i} x_{ij} * R_{ij}^{hw} \right) * \left(\sum_{j=1}^{q_i} y_{ij} * R_{ij}^{sw} \right) \right)$$

Необходимо найти такой набор $\{x_{ij}, y_{ij}\}$, чтобы R_{System} была максимальной при ограничении $C_{System} < C_{System}^{max}$

Схема алгоритма (ограниченный перебор):

1. Берём 1й модуль и анализируем его версии аппаратных и программных компонент следующим образом: для всех аппаратных и программных версий анализируем стоимость и надёжность. Отбрасываем все версии с меньшей надёжностью при большей или равной стоимости.
2. Повторяем шаг 1 для модулей 2..N.
3. Для оставшихся версий действуем следующим образом. Генерируем конфигурацию, в которой берутся версии с наименьшим номером из оставшихся вариантов.
4. Оценивается стоимость системы C_{System} .
5. Если $C_{System} > C_{System}^{max}$, то пропускаем шаги 6 – 7.
6. Оценивается надёжность системы R_{System} .
7. Если $C_{System} \leq C_{System}^{max}$ и R_{System} больше текущего лучшего решения (или текущая конфигурация становится лучшим решением при его отсутствии), то текущая конфигурация становится лучшим решением.
8. Шаги 4 – 7 повторяются для всех возможных сочетаний версий программных и аппаратных компонентов во всех модулях. Порядок обхода определяется самостоятельно.
9. Принять выбранное лучшее решение в качестве ответа.

Формат входного XML-файла:

XML-файл с описанием ВС должен иметь следующую структуру (пример файла в приложении):

- `<system>` - корневой элемент в описании ВС; имеет атрибут *limitcost* - максимально допустимая стоимость ВС; содержит теги `<module>`;
- `<module>` - описание модуля ВС; имеет атрибут *num* - номер, содержит теги `<hw>`, `<sw>`;

- `<hw>`, `<sw>` - описание доступных аппаратных и программных компонентов; имеют атрибуты `num` - номер компонента, `cost` - стоимость, `rel` - надежность;

Формат выходного XML-файла:

XML-файл с результатом выбора конфигурации ВС должен иметь следующую структуру:

- `<system>` - корневой элемент в описании ВС; имеет атрибуты `limitcost` - максимально допустимая стоимость ВС, `rel` - надежность выбранной конфигурации ВС, `cost` - стоимость выбранной конфигурации ВС, `iteration` - количество итераций алгоритма для получения решения;
- `<module>` - описание модуля ВС; имеет атрибут `num` - номер, содержит теги `<hw>`, `<sw>`;
- `<hw>`, `<sw>` - описание выбранных аппаратных и программных компонентов; имеют атрибуты `num` - номер компонента, `cost` - стоимость, `rel` - надежность;

Требуется сделать (обязательная часть):

1. Реализовать предложенный алгоритм.
2. Считать данные из xml-файла.
3. Провести исследование алгоритма на предложенном файле исходных данных и создать XML-файл с результатом.
4. Оценить достоинства и недостатки алгоритма.

Требуется сделать (бонусная часть):

1. Реализовать полный перебор и сравнить с реализованным жадным алгоритмом.
2. Провести исследование на дополнительных заданных вручную или сгенерированных автоматически конфигурациях.
3. Предложить модификации жадного алгоритма, дополняющие шаги 1 – 6 и улучшающие работу алгоритма.

Что надо прислать в качестве ответа:

1. Файл(ы) с реализацией. Код должен содержать комментарии.
2. Результаты экспериментов.
3. Пояснительный текст (если требуется).

Сроки сдачи:

22.04.2015 – 23:59 (MSK) – мягкий дедлайн (для 2го курса).

28.04.2015 – 23:59 (MSK) – жёсткий дедлайн.

Пример выходного xml-файла (по исходным данным из файла-примера):

```
<system limitcost="180" rel="0.612" cost="180" iteration="123">
  <module num="1">
    <sw num="2" rel="0.908" cost="10"/>
    <hw num="2" rel="0.980" cost="10"/>
  </module>
  <module num="2">
    <sw num="3" rel="0.887" cost="10"/>
    <hw num="2" rel="0.995" cost="20"/>
  </module>
  <module num="3">
    <sw num="1" rel="0.978" cost="20"/>
    <hw num="1" rel="0.994" cost="20"/>
  </module>
  <module num="4">
    <sw num="1" rel="0.950" cost="20"/>
    <hw num="3" rel="0.985" cost="10"/>
  </module>
  <module num="5">
```

```
        <sw num="3" rel="0.967" cost="10"/>
        <hw num="2" rel="0.980" cost="20"/>
    </module>
    <module num="6">
        <sw num="1" rel="0.908" cost="10"/>
        <hw num="2" rel="0.995" cost="20"/>
    </module>
</system>
```