# Постановка задачі

Задана компанія, що займається складанням комп’ютерів. Кожне замовлення, що надходить до компанії, є набором деталей певного типу та кількості, збиранням яких займаються робітники, що мають спеціалізацію(перерахунок усіх типів деталей, що може зібрати заданий робітник). Необхідно знайти розклад виконання замовлень, що надходять до компанії, при чому після складання розкладу з замовником укладається контракт, котрий гарантує виконання даного замовлення( тобто наступні етапи знаходження розкладу, мають точно виконати всі ті замовлення, з якими укладено контракт).

Замовлення має наступні особливості:

* Порядок збору деталей є не важливим;
* З попереднього пункту випливає, що збір комп’ютера можна здійснювати паралельно;
* Збір деталі є неперервною операцією;
* Час збору деталі залежить лише від її типу;

Робітники мають наступні особливості:

* Робочий графік – з 10:00 години ранку до 18:00 години вечора;

# Опис математичної моделі знаходження розкладу виготовлення товарів

Існує *n* замовлень збору комп’ютера, кожен з яких містить наступну інформацію:

*d i* – крайній термін виконання *i*-ого замовлення ;

*Li* – множина деталей, що необхідно зібрати в *і*-ому замовленні;

| *Li* | *= m –* кількість деталей, що необхідно зібрати в *і*-ому замовленні;

Кожна деталь, в свою чергу, має наступні параметри:

*u –* тип деталі;

*lu* – час збирання деталі *u*-ого типу(задано в хвилинах);

Збирання комп’ютерів здійснює *k* робітників, при чому задано:

*cs* – зарплатня *s*-ого робітника за 1 годину роботи;

*Ks* – множина типів деталей, що може збирати даний робітник( спеціалізація робітника);

Також для спрощення розгляду задачі введемо наступні параметри:

tij – час збирання *j*-ої деталі в *i*-ому замовленні;

cs’ - зарплатня *s*-ого робітника за 1 хвилину роботи;

*i∈[1,n]; j∈[1,m]; s∈[1,k]; u∈[1,p]*

Необхідно знайти наступні параметри:

T\_bijs – час початку збору *j*-ої деталі в *i*-ому замовленні *s*–им робітником;

T\_eijs – час кінця збору *j*-ої деталі в *i*-ому замовленні *s*–им робітником;

Також для спрощення введемо наступні параметри:

– фактичний час завершення виконання замовлення після знаходження розкладу; *(1)*

Для задовільного розкладу мають виконуватись наступні обмеження:

– фактичний час завершення виконання розкладу не перевищує крайнього терміну виконання замовлення; *(2)*

Цільовою функцією є:

*–* мінімізація витрат на заробітну плату робітників; *(3)*

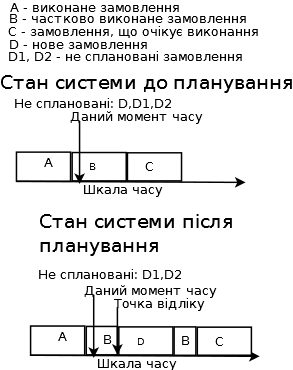
# Алгоритм рішення задачі знаходження розкладу виготовлення товарів

## Уточнення та спрощення математичної моделі

Розглянувши постановку задачі очевидно, що дана система є динамічною, тобто в будь-який момент часу може прийти будь-яка кількість замовлень. Складання таких розкладів є вкрай складною задачею, тому перетворимо умову задачі наступним чином:

* Початок складання розкладу буде здійснено лише для одного замовлення, виконання якого ще не є спланованим. Надалі – нове замовлення;
* Час планування визначається як 10:00 година ранку наступного дня від початку складання розкладу. Надалі введемо термін – «точка відліку»;
* «Точка відліку» поділяє всі замовлення, що знаходяться в компанії, на 3 категорії: виконані(завершені) замовлення, не виконані(нове замовлення, частково завершені замовлення та замовлення, що очікують виконання) і не сплановані(замовлення, розгляд яких ще не відбувся);
* Складання нового розкладу буде відбуватися для всіх замовлень другої категорії;

Дані умови показані на наступному рисунку:



Таким чином, отримали статичний характер системи – в певний момент часу знаходження розкладу буде відбуватися для відомих та визначених замовлень.

## Узагальнений алгоритм складання розкладу виготовлення товарів

На основі отриманої інформації складемо узагальнений алгоритм знаходження розкладу виготовлення комп’ютерів:

1. Знайдемо всі замовлення 2 категорії та запишемо їх до множини пошуку робіт(U)
2. Поки U не порожня: виберемо з U найкращу роботу та вилучимо її з множини пошуку
   1. Знайдемо всі не виконані деталі(частково виконані замовлення, містять в собі об’єм робіт, що вже був зроблений) в даній роботі та запишемо їх до множини пошуку операцій(V)
   2. Поки V не порожня: виберемо з V найкращу операцію та вилучимо її з множини пошуку
      1. Серед множини доступних робітників для даної операції виберемо найкращого робітника(X)
      2. До шкали робочого процесу X запишемо дану операцію
      3. Зі шкали робочого процесу X отримаємо T\_bijs та T\_eijs та запишемо до множини-результату(R)
3. Перевіримо чи виконується для R умова *(2)*

## Введення критеріїв. Створення альтернативних алгоритмів.

Цільовою функцією для нашої моделі є отримання якомога «дешевого» розкладу, але при цьому не враховується ряд інших важливих факторів(серед яких, наприклад, – дотримання директивних термінів). Даний аспект вирішимо наступним чином: в узагальненому алгоритмі не визначено як саме відбувається пошук найкращої роботи(операції, робітника), тому можна скласти критерії пошуку, котрі будуть покращувати кінцевий розклад за своїми визначеними локальними параметрами(наприклад, мінімізація максимального запізнення виконання замовлень) та комбінувати їх для створення альтернативних розкладів, а глобальний критерій(цільова функція моделі) буде вибирати серед них найвигідніший.

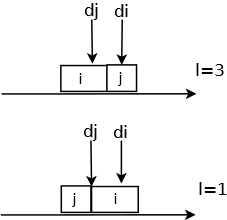
## Опис критеріїв

1. *Критерій вибору роботи*
   1. *Вибір роботи за зростанням директивного терміну(Автор: Кмець Максим)*

Даний критерій впорядковує роботи наступним чином: робота *i* передує роботі *j*, якщо *di<dj*. Таким чином забезпечується мінімізація максимального запізнення, що є ідеальним критерієм для сортування робіт, тому що задовільним вважається такий розклад, котрий виконує всі розклади вчасно. А отже, максимальне запізнення у такому розкладі рівне 0, тобто очевидно, що якщо скласти задовільний розклад не можливо в будь-якому випадку, то величина запізнення( максимальна, сумарна, середня) не грає ніякої ролі, але якщо розклад можна скласти задовільним, то мінімізація максимального запізнення знайде такий розклад гарантовано.

Доведення, що перестановка двох робіт за зростанням директивного терміну зменшує максимальний час запізнення:

Не порушуючи загальності припустимо, що в певному розкладі лише 2 роботи *i* та *j* не є впорядкованими за директивним терміном( *i* передує роботі *j*, але *dj<di*) і максимальне запізнення дорівнює *l*(очевидно, що відповідає запізненню роботи j). Тоді при перестановці *i* та *j* місцями, запізнення для *j* зменшиться, а запізнення для *i* буде на *di-dj* меншим від старого запізнення для *j.* Таким чином максимальне запізнення стане менше.

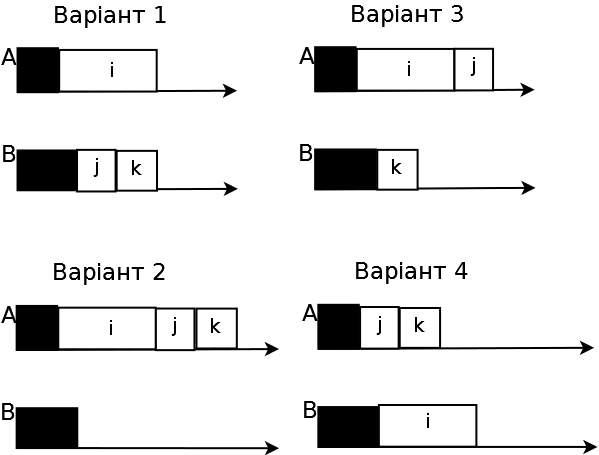


1. *Критерій вибору операції*

Критерії вибору операції є тісно пов’язаними з вибором робітника, що виконує дану операцію, тому представлені нижче критерії базуються на пункт 3 даного розділу

* 1. *Вибір операції за спаданням тривалості її виконання(Автор: Тимчук Андрій)*

Впорядкування для даного критерію відбувається так: операція *i* передує операції *j*(в даному пункті приналежність до замовлення немає значення), якщо *ti>tj*. Суть критерію полягає в наступному: нехай існує два робітника A та B та три операції *i,j,k* з тривалістю *ti > tj, ti > tk, tj~tk*. Можливі наступні варіанти розподілу операцій:

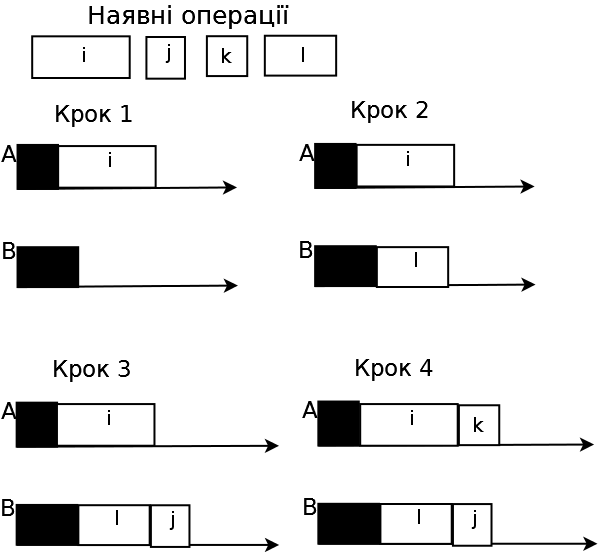


Варіант 1 відповідає найкращому розподіленню операцій між робітниками і можливий лише за умови призначення найдовшої операції найбільш незайнятому робітнику( варіанти 2 та 3 надмірно навантажують робітника A, а варіант 4 призначає найдовшу операцію більш зайнятому робітнику. Є гіршим за 1 варіант як мінімум на початкову різницю зайнятості робітників A та B ).

* 1. *Вибір операції за фактором завантаженості спеціалізації(Автор: Вереня Олександр)*

1. *Критерій вибору робітника*
   1. *Вибір найменш зайнятого робітника(Автор: Андрій Тимчук)*

Для даного класу задач важливою властивістю є щільність розкладу, тобто максимальна завантаженість усіх робітників підприємства. Забезпечити таку властивість допомагає наступний критерій: операція *i* призначається такому з робітників *A* та *B(робітники можуть виконати операцію i),* у якого *cA<cB*, де *c* – функція зайнятості. Отже, на кожній ітерації робітники будуть рівномірно розподіляти операції між собою, забезпечуючи сумарне зменшення тривалості виконання замовлення.



* 1. *Вибір найбільш дешевого робітника(Автор: Вереня Олександр)*

Цей критерій є модифікацією попереднього, а саме: якщо два робітника A та B є найбільш незайнятими, то вибір відбувається на основі погодинної заробітної плати, тобто вибирається «дешевший» робітник, що дозволяє покращити розклад з точки зору цільової функції вихідної математичної моделі.

## Опис та дослідження альтернативних алгоритмів

1. *Алгоритм «Білий ведмідь». <1.1; 2.1; 3.1>( Досліджував: Кмець Максим)*
2. *Алгоритм «Дрізд чорний». <1.1; 2.2; 3.1>( Досліджував: Вереня Олександр)*
3. *Алгоритм «Королівський пінгвін». <1.1; 2.2; 3.2>( Досліджував: Андрій Тимчук)*