

Приклад. Структури, об'єднання та перерахування

```
each: function(e, t, n) {
  var r, i = 0,
      o = e.length,
      u = H(e);
  if (n) {
    if (n) {
      for (; o > i; i++)
        if (r = t.apply(e[i], n), r !== !1) break
    } else
      for (i in e)
        if (r = t.apply(e[i], n), r !== !1) break
  } else if (a) {
    for (; o > i; i++)
      if (r = t.call(e[i], i, e[i]), r !== !1) break
  } else
    for (i in e)
      if (r = t.call(e[i], i, e[i]), r !== !1) break;
  return e
},
trim: b && b.call("\uffeff\u00a0") ? function(e) {
  return null == e ? "" : b.call(e)
} : function(e) {
  return null == e ? "" : (e + "").replace(C, "")
},
makeArray: function(e, t) {
  var n = t || [];
  return null != e && (H(Object(e)) ? x.merge(n, "string" == typeof e ? [e] : e) : b.call(
),
isArray: function(e, t, n) {
  var r, i;
  if (t)
    if (n) return n.call(t, e, n);
    for (r = e.length, n = n ? 0 > n ? Math.max(0, r + n) : 0 : 0; r > n; n++)
      if (n in t && t[n] === e) return n
}
```





Каталог товарів.

Створити структуру **Товари (Prod)** з полями: **name**, **price**, **category** (використати *enum* — наприклад, - їжа , електротовари, одяг) та **union** для зберігання ваги (в кг), потужності (в Вт) або розмір, в залежності від категорії.

Ввести дані про товари, і реалізувати пошук найдешевших товарі у кожній категорії.

Код завдання



```
#include <stdio.h>
#include <string.h>
```

```
#define N 7 //Кількість товарів можна задавати довільно
```

Оголошення основних типів даних

```
typedef enum {
    food,
    electronics,
    clothing
} Category;
```

```
typedef union {
    float weight;
    int size;
    int power;
} ProdData;
```

```
typedef struct {
    char name[50];
    float price;
    Category cg;
    ProdData data;
} Prod;
```



```
void printProd(Prod p) {
    printf("Назва: %s\n", p.name);
    printf("Ціна: %.2f грн\n", p.price);
    printf("Категорія: ");
    switch(p.cg) {
        case food:
            printf("Їжа\n");
            printf("Вага: %.2f кг\n", p.data.weight);
            break;
        case electronics:
            printf("Електротовари\n");
            printf("Потужність: %d Вт\n", p.data.power);
            break;
        case clothing:
            printf("Одяг\n");
            printf("Розмір: %d\n", p.data.size);
            break;
    }
    printf("-----\n");
}
```

**Головна програма
починається з
вибору варіанту
вводу**

**Формування власної
функції для виводу**

```
int main() {
    Prod prods[N];

    int choice;
    printf("Оберіть спосіб заповнення товарів:\n");
    printf("1 - Ввести вручну\n");
    printf("2 - Використати готовий список\n");
    printf("Ваш вибір: ");
    scanf("%d", &choice);
}
```

Формування блоку вводу даних:



```
if (choice == 1) {
    printf("\nВведіть дані про %d товарів:\n", N);
    for (int i = 0; i < N; i++) {
        printf("\nТовар #%d\n", i + 1);
        printf("Назва: ");
        getchar();
        fgets(prods[i].name, sizeof(prods[i].name), stdin);
        prods[i].name[strcspn(prods[i].name, "\n")] = '\0';

        printf("Ціна (грн): ");
        scanf("%f", &prods[i].price);

        int cat;
        printf("Категорія (0 - їжа, 1 - електротовари, 2 - одяг): ");
        scanf("%d", &cat);
        prods[i].cg = (Category)cat;

        if (prods[i].cg == electronics) {
            printf("Потужність (Вт): ");
            scanf("%d", &prods[i].data.power);
        } else if (prods[i].cg == food) {
            printf("Вага (кг): ");
            scanf("%f", &prods[i].data.weight);
        } else {
            printf("Розмір: ");
            scanf("%d", &prods[i].data.size);
        }
    }
}
```

Формування блоку вводу даних



```
} else {
    Prod temp[N] = {
        {"Холодильник LG", 12000.0, electronics, .data.power = 300},
        {"Молоко", 50.0, food, .data.weight = 1.0},
        {"Куртка зимова", 1500.0, clothing, .data.size = 44},
        {"Телевізор Samsung", 8500.0, electronics, .data.power = 250},
        {"Сир", 220.0, food, .data.weight = 0.5},
        {"Ноутбук HP", 21000.0, electronics, .data.power = 90},
        {"Футболка", 300.0, clothing, .data.size = 48}
    };
    memcpy(prods, temp, sizeof(temp));
    printf("\nБуло використано готовий список товарів.\n");
}
```

Формування блоку пошуку найдешевших товарів у категоріях



```
for (int i = 0; i < N; i++) {
    switch (prods[i].cg) {
        case food:
            if (minFdPrice == -1 || prods[i].price < minFdPrice) {
                minFdPrice = prods[i].price;
                minFdIndex = i;
            }
            break;
        case electronics:
            if (minElPrice == -1 || prods[i].price < minElPrice) {
                minElPrice = prods[i].price;
                minElIndex = i;
            }
            break;
        case clothing:
            if (minClPrice == -1 || prods[i].price < minClPrice) {
                minClPrice = prods[i].price;
                minClIndex = i;
            }
            break;
    }
}
```

Виведення отриманих результатів



```
printf("\n=== Найдешевші товари у кожній категорії ===\n");

if (minFdIndex != -1) {
    printf("\nКатегорія: Їжа\n");
    printProd(prods[minFdIndex]);
} else {
    printf("\nУ категорії Їжа немає товарів.\n");
}

if (minElIndex != -1) {
    printf("\nКатегорія: Електротовари\n");
    printProd(prods[minElIndex]);
} else {
    printf("\nУ категорії Електротовари немає товарів.\n");
}

if (minClIndex != -1) {
    printf("\nКатегорія: Одяг\n");
    printProd(prods[minClIndex]);
} else {
    printf("\nУ категорії Одяг немає товарів.\n");
}

return 0;
}
```

Результат виконання



```
Оберіть спосіб заповнення товарів:
1 - Ввести вручну
2 - Використати готовий список
Ваш вибір: 2

Було використано готовий список товарів.

=== Найдешевші товари у кожній категорії ===

Категорія: Їжа
Назва: Молоко
Ціна: 50.00 грн
Категорія: Їжа
Вага: 1.00 кг
-----

Категорія: Електротовари
Назва: Телевізор Samsung
Ціна: 8500.00 грн
Категорія: Електротовари
Потужність: 250 Вт
-----

Категорія: Одяг
Назва: Футболка
Ціна: 300.00 грн
Категорія: Одяг
Розмір: 48
-----

...Program finished with exit code 0
Press ENTER to exit console.
```

```
Оберіть спосіб заповнення товарів:
1 - Ввести вручну
2 - Використати готовий список
Ваш вибір: 1

Введіть дані про 7 товарів:

Товар #1
Назва: Пальто
Ціна (грн): 3050
Категорія (0 - їжа, 1 - електротовари, 2 - одяг): 2
Розмір: 48

Товар #2
Назва: Сир
Ціна (грн): 240
Категорія (0 - їжа, 1 - електротовари, 2 - одяг): 0
Вага (кг): 1

Товар #3
Назва: Молоко
Ціна (грн): 65
Категорія (0 - їжа, 1 - електротовари, 2 - одяг): 0
Вага (кг): 1

Товар #4
Назва: Хліб
Ціна (грн): 32
Категорія (0 - їжа, 1 - електротовари, 2 - одяг): 0
Вага (кг): .7

Товар #5
Назва: Футболка
Ціна (грн): 300
Категорія (0 - їжа, 1 - електротовари, 2 - одяг): 2
Розмір: 44

Товар #6
Назва: Піца
Ціна (грн): 650
Категорія (0 - їжа, 1 - електротовари, 2 - одяг): 0
Вага (кг): 1

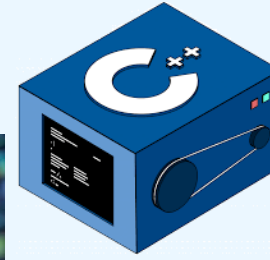
Товар #7
Назва: Брюки
Ціна (грн): 1500
Категорія (0 - їжа, 1 - електротовари, 2 - одяг): 2
Розмір: 48

=== Найдешевші товари у кожній категорії ===

Категорія: Їжа
Назва: Хліб
Ціна: 32.00 грн
Категорія: Їжа
Вага: 0.70 кг
-----

У категорії Електротовари немає товарів.

Категорія: Одяг
Назва: Футболка
Ціна: 300.00 грн
Категорія: Одяг
Розмір: 44
-----
```



```
each: function(o, n) {
  var i, l = 0;
  m = o.length;
  n = n || 1;
  if (n) {
    if (n > 1) {
      for (; i < m; i++)
        if (r = t.apply(o[i], n), r === !1) break
    } else
      for (i in o)
        if (r = t.apply(o[i], n), r === !1) break
    } else if (o) {
    for (; o > 1; i++)
      if (r = t.call(o[i], i, o[i]), r === !1) break
    } else
      for (i in o)
        if (r = t.call(o[i], i, o[i]), r === !1) break;
    return o
  },
  trim: b && !b.call("u0eff\u0000") ? function(e) {
  return null == e ? "" : b.call(e)
} : function(e) {
  return null == e ? "" : (e + "").replace(C, "")
},
mergeArray: function(e, t) {
  var n = t || [];
  return null != e && (Object(e) ? x.merge(n, "string" == typeof e ? [e] : e) : b.call(n, e))
},
isArray: function(e, t, n) {
  var r;
  if (!e)
    return !1;
  if (n)
    return b.call(t, e, n);
  for (r = 0; r < e.length; r++)
    if (r > 0 && e[r] > n ? Math.max(0, r + n) : n < 0; r > n; r++)
      if (n < 0 && e[r] === n)
        return !0
  return !1
}
```

Дякую за увагу!