

Практическое задание 3. Git

Оценка 3:

1. Используйте git status, чтобы узнать, на какой ветке вы находитесь.

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
```

2. Как выглядит git log?

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git log
commit 79375b0ea995d59657f936a36b38db127ffbf7bc (HEAD -> main, origin/main,
origin/HEAD)
Author: Segoga <129649003+segoga@users.noreply.github.com>
Date:   Mon Mar 3 15:43:32 2025 +0700

    Rename 1_1.c to 1.c

commit 30ecf4659f0c302e3ecba8c400fc157c001f2925
Author: Segoga <129649003+segoga@users.noreply.github.com>
commit 79375b0ea995d59657f936a36b38db127ffbf7bc (HEAD -> main, origin/main,
origin/HEAD)
Author: Segoga <129649003+segoga@users.noreply.github.com>
Date:   Mon Mar 3 15:43:32 2025 +0700
```

3. Создайте файл sort.c и вставьте туда код функции любой сортировки (только ф-ия сортировки)

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ cat sort.c
void SelectSort(struct Student students[N]){
    for (int i=0;i<N;i++){
        int k = i;
        for (int j=i+1;j<N;j++){
            if (students[k].sum < students[j].sum){
                k=j;}}
    struct Student stud=students[i];
    students[i]=students[k];
    students[k]=stud;}
}
```

4. Как сейчас выглядит вывод git status?

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  ./

nothing added to commit but untracked files present (use "git add" to track)
```

5. Добавьте файл в область stage (add)

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git add sort.c
```

6. Как сейчас выглядит git status?

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   sort.c
```

7. Закоммитить файл в репозиторий

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git commit -m "Новый файл"
[main 4a4da0d] Новый файл
 1 file changed, 10 insertions(+)
 create mode 100644 lab3/sort.c
```

8. Как сейчас выглядит git status?

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
```

9. Добавить комментарий с любым текстом в этот же файл

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ cat sort.c
//qwerty

void SelectSort(struct Student students[N]){
    for (int i=0;i<N;i++){
        int k = i;
        for (int j=i+1;j<N;j++){
            if (students[k].sum < students[j].sum){
                k=j;}}
        struct Student stud=students[i];
        students[i]=students[k];
        students[k]=stud;}
}
```

10. Как сейчас выглядит git status?

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   sort.c

no changes added to commit (use "git add" and/or "git commit -a")
```

11. Добавьте (add) изменение файла

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git add sort.c
```

12. Как сейчас выглядит git status?

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   sort.c
```

13. Измените файл еще раз (можно добавить еще комментарий или убрать старый)

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ cat sort.c
//qwerty
//qwerty2.0
void SelectSort(struct Student students[N]){
    for (int i=0;i<N;i++){
        int k = i;
        for (int j=i+1;j<N;j++){
            if (students[k].sum < students[j].sum){
                k=j;}}
    struct Student stud=students[i];
    students[i]=students[k];
    students[k]=stud;}
}

```

14. Сделайте коммит

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git commit -m "qwerty2.0"
[main aef3ca6] qwerty2.0
1 file changed, 2 insertions(+)

```

15. Как сейчас выглядит status? Журнал (log)?

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git status
On branch main
Your branch is ahead of 'origin/main' by 2 commits.
  (use "git push" to publish your local commits)

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   sort.c

no changes added to commit (use "git add" and/or "git commit -a")

```

```
Author: seg0ga <demin_s22@mail.ru>
Date:   Mon Mar 3 16:17:15 2025 +0700
```

qwerty2.0

```
commit 4a4da0d998ad8983d8a62f2a0b5f766af629720e
```

```
Author: seg0ga <demin_s22@mail.ru>
Date:   Mon Mar 3 16:13:50 2025 +0700
```

Новый файл

```
commit 79375b0ea995d59657f936a36b38db127ffbf7bc (origin/main, origin/HEAD)
```

```
Author: Segoga <129649003+seg0ga@users.noreply.github.com>
Date:   Mon Mar 3 15:43:32 2025 +0700
```

Rename 1_1.c to 1.c

```
commit 30ecf4659f0c302e3ecba8c400fc157c001f2925
```

```
Author: Segoga <129649003+seg0ga@users.noreply.github.com>
Date:   Mon Mar 3 15:43:12 2025 +0700
```

Create 2.c

```
commit 7c4cde851f7c0f4ce7775856ef60ab4994cc6eb2
```

```
Author: Segoga <129649003+seg0ga@users.noreply.github.com>
```

16. Добавьте в stage и закоммитьте последнее изменение

```
segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git add sort.c
segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git commit -m "qwerty3.0"
[main 5018656] qwerty3.0
1 file changed, 1 insertion(+), 1 deletion(-)
```

17. Запустим на удаленный репо (git push)

```
segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ git push
Username for 'https://github.com': seg0ga
Password for 'https://seg0ga@github.com':
remote: Invalid username or password.
fatal: Authentication failed for 'https://github.com
/seg0ga/ProgrammingC_2_sibsutis/'
```

Теперь мы немного поиграем с ветками.

1. Используйте git branch mybranch (или git checkout -b mybranch), чтобы создать новую ветку с именем mybranch.

```
segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ git branch mybran
ch
```

2. Снова используйте `git branch`, чтобы увидеть новую созданную ветку.

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ git branch
* main
  mybranch
```

3. Используйте `git switch mybranch` (или `git checkout mybranch`), чтобы переключиться на новую ветку.

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ git switch mybran
ch
M      lab3/sort2.c
Switched to branch 'mybranch'
```

4. Как изменяется вывод `git status` при переключении между `master` и новой веткой, которую вы создали?

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ git status
On branch mybranch
Changes not staged for commit:
  (use "git add <file>..." to update what will be co
mmitted)
  (use "git restore <file>..." to discard changes in
working directory)
        modified:   sort2.c

no changes added to commit (use "git add" and/or "gi
t commit -a")

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be co
mmitted)
  (use "git restore <file>..." to discard changes in
working directory)
        modified:   sort2.c

no changes added to commit (use "git add" and/or "gi
t commit -a")
```

5. Убедитесь, что вы находитесь на своей ветке `mybranch`, прежде чем продолжить.


```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ git switch mybran
ch
M      lab3/sort2.c
Switched to branch 'mybranch'
```

6. Создайте файл с именем file1.txt и своим именем.

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ nano file1.txt
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ cat file1.txt
Sergei
```

7. Добавьте файл и закоммитьте это изменение.

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ git add file1.txt
| git commit -m "QWERTY"
[mybranch f5f4d8b] QWERTY
1 file changed, 1 insertion(+)
create mode 100644 lab3/file1.txt
```

8. Используйте git log --oneline --graph, чтобы увидеть, что ваша ветка указывает на НОВЫЙ КОММИТ.

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ git log --oneline
--graph
* f5f4d8b (HEAD -> mybranch) QWERTY
* 1b18dbe (origin/main, origin/HEAD, main) qwe
* 5018656 qwerty3.0
* aef3ca6 qwerty2.0
* 4a4da0d Новый файл
* 79375b0 Rename 1_1.c to 1.c
* 30ecf46 Create 2.c
* 7c4cde8 Update README.md
* 3f9d697 Update README.MD
* 0c204fd Update README.MD
* 3dae471 Update README.MD
* 45625a1 Update README.MD
* 207e6de Update README.MD
* 88e4e77 Update README.md
* 08ed5dc Update README.md
* e7169a8 Create README.md
```

9. Вернитесь к ветке с именем master.

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ git switch main
M      lab3/sort2.c
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
```

10. Используйте git log --oneline --graph, что изменилось?

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ git log --oneline
--graph
* 1b18dbe (HEAD -> main, origin/main, origin/HEAD) q
we
* 5018656 qwerty3.0
* aef3ca6 qwerty2.0
* 4a4da0d Новый файл
* 79375b0 Rename 1_1.c to 1.c
* 30ecf46 Create 2.c
* 7c4cde8 Update README.md
* 3f9d697 Update README.MD
* 0c204fd Update README.MD
* 3dae471 Update README.MD
* 45625a1 Update README.MD
* 207e6de Update README.MD
* 88e4e77 Update README.md
* 08ed5dc Update README.md
* e7169a8 Create README.md
* 80dbcb1 Update README.md
* 64c5395 Update README.md
* feb2432 Update 1_1.c
* 5d6690e Update README.MD
* 0b15131 Delete lab1/ЗАДАНИЕ
* c60011e Create README.MD

```

11. Создайте новый файл с именем file2.txt и закоммитьте его.

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ git add file2.txt
| git commit -m "abcdef"
fatal: pathspec 'file2.txt' did not match any files
error: pathspec '-' did not match any file(s) known
to git
error: pathspec 'm' did not match any file(s) known
to git
error: pathspec 'abcdef' did not match any file(s) k
nown to git

```

12. Используйте `git log --oneline --graph --all`, чтобы увидеть, что ваша ветка указывает на новый коммит, и что теперь у двух веток разные коммиты.


```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ git log --oneline
--graph --all
* f5f4d8b (mybranch) QWERTY
* 1b18dbe (HEAD -> main, origin/main, origin/HEAD) q
we
* 5018656 qwerty3.0
* aef3ca6 qwerty2.0
* 4a4da0d Новый файл
* 79375b0 Rename 1_1.c to 1.c
* 30ecf46 Create 2.c
* 7c4cde8 Update README.md
* 3f9d697 Update README.MD
* 0c204fd Update README.MD
* 3dae471 Update README.MD
* 45625a1 Update README.MD
* 207e6de Update README.MD
* 88e4e77 Update README.md
* 08ed5dc Update README.md
* e7169a8 Create README.md
* 80dbcb1 Update README.md
* 64c5395 Update README.md
* feb2432 Update 1_1.c
* 5d6690e Update README.MD
* 0b15131 Delete lab1/ЗАДАНИЕ
* c69011c Create README.MD
* 6f04c60 Rename sort.c to 1_1.c
* e9a11b4 Update sort.c
* 58f0967 Rename ЗАДАНИ to ЗАДАНИЕ
* b58c0f9 Update README.md
* 9f5759c Create README.md
* 0a03bf6 Create ЗАДАНИЕ
* e7daf23 Create sort.c

```

13.Переключитесь на вашу ветку mybranch.

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ git switch mybran
ch
M      lab3/sort2.c
Switched to branch 'mybranch'

```

14.Наш text2.txt пропал?

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ ls
file1.txt  sort.c  sort2.c  text2.txt

```

15.Используйте git diff mybranch master, чтобы увидеть разницу между двумя ветками.

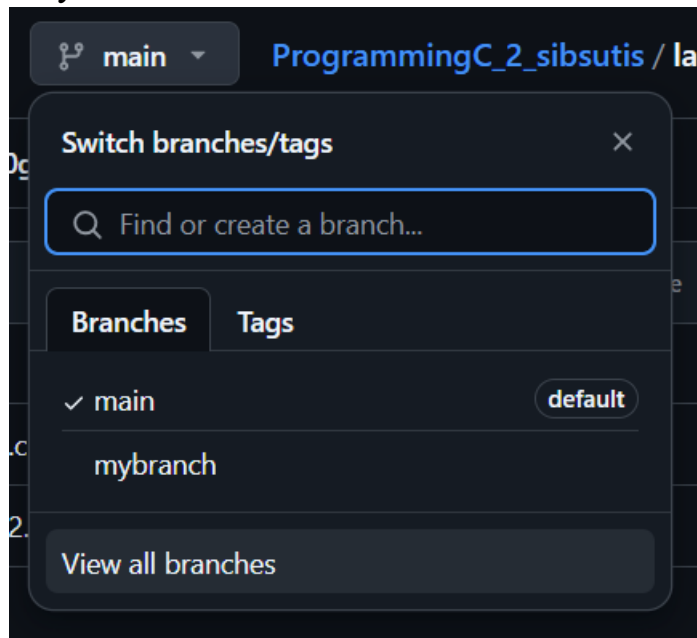
```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab
_git/ProgrammingC_2_sibsutis/lab3$ git diff mybranch
main
diff --git a/lab3/file1.txt b/lab3/file1.txt
deleted file mode 100644
index 97fd905..0000000
--- a/lab3/file1.txt
+++ /dev/null
@@ -1,0,0 @@
-Sergei
```

16. Добавить текстовый документ со скриншотами в ветку **mybranch**.

Закоммитить и запустить на удаленный репо ветку mybranch (git push -u origin mybranch)

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibsutis/lab3$ git push -u origin mybranch
Username for 'https://github.com': segoga
Password for 'https://segoga@github.com':
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 20 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (9/9), 700 bytes | 35.00 KiB/s, done.
Total 9 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), completed with 1 local object.
remote:
remote: Create a pull request for 'mybranch' on GitHub by visiting:
remote:   https://github.com/segoga/ProgrammingC_2_sibsutis/pull/new/mybranch
remote:
To https://github.com/segoga/ProgrammingC_2_sibsutis
 * [new branch]      mybranch -> mybranch
Branch 'mybranch' set up to track remote branch 'mybranch' from 'origin'.
```

17. Убедиться что в github.com две ветки master и mybranch. Не забыть запустить изменения master ветки в master



Оценка 4:

Выполнить на оценку 3, а затем..

1. Переключитесь на ветку mybranch. В ней будет файл sort.c из предыдущих шагов с функцией сортировки

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs  
utis/lab3$ git switch mybranch  
M      lab3/sort2.c  
A      lab3/text2.txt  
Already on 'mybranch'  
Your branch is up to date with 'origin/mybranch'.
```

2. Перезапишите содержимое в sort.c добавив функцию main(), в которой будет объявлен массив из нескольких чисел (пример `int a[] = {4, 2, 0};`) и вызвана функция сортировки для этого массива.

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs  
utis/lab3$ cat sort.c  
#include <stdio.h>  
  
void bubble_sort(int arr[], int n) {  
    for (int i = 0; i < n-1; i++) {  
        for (int j = 0; j < n-i-1; j++) {  
            if (arr[j] > arr[j+1]) {  
                int temp = arr[j];  
                arr[j] = arr[j+1];  
                arr[j+1] = temp;}}}}  
  
int main() {  
    int a[] = {4, 2, 0};  
    int n = sizeof(a) / sizeof(a[0]);  
  
    printf("Исходный массив: ");  
    for (int i = 0; i < n; i++) {  
        printf("%d ", a[i]);  
    }  
    printf("\n");  
  
    bubble_sort(a, n);  
  
    printf("Отсортированный массив: ");  
    for (int i = 0; i < n; i++) {  
        printf("%d ", a[i]);  
    }
```

3. Что вам говорит git diff?

```

diff --git a/lab3/sort.c b/lab3/sort.c
index 12fe397..b2ccb01 100644
--- a/lab3/sort.c
+++ b/lab3/sort.c
@@ -1,12 +1,28 @@
-//qwerty
-//qwerty2.0
-void SelectSort(struct Student students[N]){
-    for (int i=0;i<N;i++){
-        int k = i;
-        for (int j=i+1;j<N;j++){
-            if (students[k].sum < students[j].sum){
-                k=j;}}
-    struct Student stud=students[i];
-    students[i]=students[k];
-    students[k]=stud;}
+
+#include <stdio.h>
+
+void bubble_sort(int arr[], int n) {
+    for (int i = 0; i < n-1; i++) {
+        for (int j = 0; j < n-i-1; j++) {
+            if (arr[j] > arr[j+1]) {
+                int temp = arr[j];
+                arr[j] = arr[j+1];
+                arr[j+1] = temp;}}}}
+
+int main() {
+    int a[] = {4, 2, 0};
+    int n = sizeof(a) / sizeof(a[0]);
+
+    printf("Исходный массив: ");
+    for (int i = 0; i < n; i++) {
+

```

4. Что вам говорит git diff --staged? Пустой(нет)?

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git diff --staged
diff --git a/lab3/sort2.c b/lab3/sort2.c
index bc6fb22..aa2bc9a 100644
--- a/lab3/sort2.c
+++ b/lab3/sort2.c
@@ -1,1 @@
-qwertyu
+fdsgfdhghjqwertyu
diff --git a/lab3/text2.txt b/lab3/text2.txt
new file mode 100644
index 0000000..172d50c
--- /dev/null
+++ b/lab3/text2.txt
@@ -0,0 +1,3 @@
+
+qwerty
+

```

5. Добавьте в staged файл sort.c

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git add sort.c

```

6. Что вам говорит git diff?

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git diff

```

7. Что вам говорит git diff --staged?

```

diff --git a/lab3/sort.c b/lab3/sort.c
index 12fe397..b2ccb01 100644
--- a/lab3/sort.c
+++ b/lab3/sort.c
@@ -1,12 +1,28 @@
-//qwerty
-//qwerty2.0
-void SelectSort(struct Student students[N]){
-    for (int i=0;i<N;i++){
-        int k = i;
-        for (int j=i+1;j<N;j++){
-            if (students[k].sum < students[j].sum){
-                k=j;}}
-    struct Student stud=students[i];
-    students[i]=students[k];
-    students[k]=stud;}
+
+#include <stdio.h>
+
+void bubble_sort(int arr[], int n) {
+    for (int i = 0; i < n-1; i++) {
+        for (int j = 0; j < n-i-1; j++) {
+            if (arr[j] > arr[j+1]) {
+                int temp = arr[j];
+                arr[j] = arr[j+1];
+                arr[j+1] = temp;}}}}
+
+int main() {
+    int a[] = {4, 2, 0};
+    int n = sizeof(a) / sizeof(a[0]);
+
+    printf("Исходный массив: ");
+    for (int i = 0; i < n; i++) {
+

```

8. Удалите любое из чисел в массиве в sort.c:


```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ cat sort.c
#include <stdio.h>

void bubble_sort(int arr[], int n) {
    for (int i = 0; i < n-1; i++) {
        for (int j = 0; j < n-i-1; j++) {
            if (arr[j] > arr[j+1]) {
                int temp = arr[j];
                arr[j] = arr[j+1];
                arr[j+1] = temp;}}}}

int main() {
    int a[] = {4, 0};
    int n = sizeof(a) / sizeof(a[0]);

    printf("Исходный массив: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", a[i]);
    }
    printf("\n");

    bubble_sort(a, n);

    printf("Отсортированный массив: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", a[i]);
    }
    printf("\n");
}

```

9. Что вам говорит git diff?

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git diff
diff --git a/lab3/sort.c b/lab3/sort.c
index b2ccb01..addb61a 100644
--- a/lab3/sort.c
+++ b/lab3/sort.c
@@ -9,7 +9,7 @@ void bubble_sort(int arr[], int n) {
     arr[j+1] = temp;}}}}

int main() {
-    int a[] = {4, 2, 0};
+    int a[] = {4, 0};
    int n = sizeof(a) / sizeof(a[0]);

    printf("Исходный массив: ");

```

10. Что вам говорит git diff --staged?

```

diff --git a/lab3/sort.c b/lab3/sort.c
index 12fe397..b2ccb01 100644
--- a/lab3/sort.c
+++ b/lab3/sort.c
@@ -1,12 +1,28 @@
-//qwerty
-//qwerty2.0
-void SelectSort(struct Student students[N]){
-    for (int i=0;i<N;i++){
-        int k = i;
-        for (int j=i+1;j<N;j++){
-            if (students[k].sum < students[j].sum){
-                k=j;}}
-    struct Student stud=students[i];
-    students[i]=students[k];
-    students[k]=stud;}
+#include <stdio.h>
+
+void bubble_sort(int arr[], int n) {
+    for (int i = 0; i < n-1; i++) {
+        for (int j = 0; j < n-i-1; j++) {
+            if (arr[j] > arr[j+1]) {
+                int temp = arr[j];
+                arr[j] = arr[j+1];
+                arr[j+1] = temp;}}}}
+
+int main() {
+    int a[] = {4, 2, 0};
+    int n = sizeof(a) / sizeof(a[0]);
+
+    printf("Исходный массив: ");
+    for (int i = 0; i < n; i++) {
+

```

11. Объясните, что происходит (git diff показывает изменения в рабочей директории, которые будут добавлены в индекс, а git diff –staged показывает изменения, которые уже добавлены в индекс и будут закоммичены)
12. Запустите git status и обратите внимание, что sort.c присутствует дважды в выводе.

```

segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git status
On branch mybranch
Your branch is up to date with 'origin/mybranch'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   sort.c
        modified:   sort2.c
        new file:   text2.txt

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   sort.c

```

13. Запустите `git restore --staged sort.c`, чтобы отменить индексацию изменения

```

segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git restore --staged sort.c

```

14. Что вам теперь говорит `git status`?

```

segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git status
On branch mybranch
Your branch is up to date with 'origin/mybranch'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   sort2.c
        new file:   text2.txt

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   sort.c

```

15. Индексируйте изменение (`add`) и сделайте коммит

```

segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git add sort.c
segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git commit -m "Сделал коммит"
[mybranch efa221] Сделал коммит
3 files changed, 32 insertions(+), 13 deletions(-)
rewrite lab3/sort.c (99%)
create mode 100644 lab3/text2.txt

```

16. Как выглядит журнал?

Date: Mon Mar 3 16:51:20 2025 +0700

qwe

commit 50186569b3faac251c5008d4cf82d69c488a126f

Author: seg0ga <demin_s22@mail.ru>

Date: Mon Mar 3 16:19:06 2025 +0700

qwerty3.0

commit aef3ca6f5f74ab3f706c5d865b8d97501edc5c71

:...skipping...

commit efad2212cf65e897851b6daf176a999c99b9b286 (HEAD -> mybranch)

Author: seg0ga <demin_s22@mail.ru>

Date: Thu Mar 6 21:20:30 2025 +0700

Сделал КОММИТ

commit b4e2531650d4bae828c17719c2a191a852c6691e (origin/mybranch)

Author: seg0ga <demin_s22@mail.ru>

Date: Mon Mar 3 16:57:29 2025 +0700

Скрины

commit f5f4d8b0e239e7ad5241cab344bcb0b0304ccbd7

Author: seg0ga <demin_s22@mail.ru>

Date: Mon Mar 3 16:47:16 2025 +0700

17.Добавьте в sort.c в main() printf("hello git\n");

```
int main() {  
    printf("hello git\n")  
    int a[] = {4, 0};
```

18.Каково содержимое sort.c?

```

segoga@Notebook: /mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ cat sort.c
#include <stdio.h>

void bubble_sort(int arr[], int n) {
    for (int i = 0; i < n-1; i++) {
        for (int j = 0; j < n-i-1; j++) {
            if (arr[j] > arr[j+1]) {
                int temp = arr[j];
                arr[j] = arr[j+1];
                arr[j+1] = temp;}}}}

int main() {
    printf("hello git\n")
    int a[] = {4, 0};
    int n = sizeof(a) / sizeof(a[0]);

    printf("Исходный массив: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", a[i]);
    }
    printf("\n");

    bubble_sort(a, n);

    printf("Отсортированный массив: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", a[i]);
    }
    printf("\n");
}

```

19. Что нам говорит git status?

```

segoga@Notebook: /mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git status
On branch mybranch
Your branch is ahead of 'origin/mybranch' by 1 commit.
  (use "git push" to publish your local commits)

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   sort.c

no changes added to commit (use "git add" and/or "git commit -a")

```

20. Запустите git restore sort.c

```

segoga@Notebook: /mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git restore sort.c

```

21. Каково содержимое sort.c?

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ cat sort.c
#include <stdio.h>

void bubble_sort(int arr[], int n) {
    for (int i = 0; i < n-1; i++) {
        for (int j = 0; j < n-i-1; j++) {
            if (arr[j] > arr[j+1]) {
                int temp = arr[j];
                arr[j] = arr[j+1];
                arr[j+1] = temp;}}}}

int main() {
    int a[] = {4, 0};
    int n = sizeof(a) / sizeof(a[0]);

    printf("Исходный массив: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", a[i]);
    }
    printf("\n");

    bubble_sort(a, n);

    printf("Отсортированный массив: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", a[i]);
    }
    printf("\n");
}

```

22. Что нам говорит git status?

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git status
On branch mybranch
Your branch is ahead of 'origin/mybranch' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

```

23. Запустить на удаленный репо ветку.


```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git push origin mybranch
Username for 'https://github.com': seg0ga
Password for 'https://seg0ga@github.com':
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 20 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (6/6), 802 bytes | 42.00 KiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/seg0ga/ProgrammingC_2_sibsutis
b4e2531..efad221 mybranch -> mybranch

```

Теперь поиграемся с ветками и ff-merge

Помните: когда вы хотите обновить ветку так, чтобы она также имела все изменения из другой ветки, используйте команду 'git merge [имя ветки]', где [имя ветки] - ветка, из которой мы хотим смержить наши изменения.

1. Создать файл greeting.txt, проиндексировать его и закоммитить с сообщением "Add file greeting.txt".

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ nano greeting.txt
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git add greeting.txt
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git commit -m "Add file greeting.txt"
[mybranch 6ac0633] Add file greeting.txt
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 lab3/greeting.txt

```

2. Добавить в этот файл слово hello, индексируем и коммитим с текстом "Add content to greeting.txt"

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ nano greeting.txt
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ cat greeting.txt
hello
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git add greeting.txt
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git commit -m "Add content to greeting.txt"
[mybranch cfffc6a] Add content to greeting.txt
1 file changed, 1 insertion(+)

```

3. Создайте ветку с именем feature/uppercase (да, feature/uppercase — это совершенно допустимое имя ветки и общепринятое соглашение).

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git branch feature/uppercase

```

4. Переключитесь на эту ветку

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git switch feature/uppercase
Switched to branch 'feature/uppercase'
```

5. Каков вывод `git status`?

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git status
On branch feature/uppercase
nothing to commit, working tree clean
```

6. Отредактируйте `greeting.txt`, чтобы он содержал приветствие в верхнем регистре (HELLO)

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ cat greeting.txt
hello
HELLO
```

7. Добавьте файл `greeting.txt` и закоммитьте

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git add greeting.txt
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git commit -m "HELLO"
[feature/uppercase 5913d27] HELLO
1 file changed, 1 insertion(+)
```

8. Каков вывод `git branch`?

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git branch
* feature/uppercase
  main
  mybranch
```

9. Каков вывод `git log --oneline --graph --all`

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git log --oneline --graph --all
error: switch `l' expects a numerical value
```

10. Переключитесь на главную ветку

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git switch main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
```

11. Используйте `cat`, чтобы увидеть содержимое файла `greetings.txt`

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ cat greeting.txt
cat: greeting.txt: No such file or directory
```

12. Сравните ветки

```

new file mode 100644
index 0000000..97fd905
--- /dev/null
+++ b/lab3/file1.txt
@@ -0,0 +1 @@
+Sergei
diff --git a/lab3/file2.txt b/lab3/file2.txt
new file mode 100644
index 0000000..a1afb10
--- /dev/null
+++ b/lab3/file2.txt
@@ -0,0 +1 @@
+qwer
diff --git a/lab3/greeting.txt b/lab3/greeting.txt
new file mode 100644
index 0000000..afecd8a
--- /dev/null
+++ b/lab3/greeting.txt
@@ -0,0 +1,2 @@
+hello
+HELLO
diff --git a/lab3/qwerty.txt b/lab3/qwerty.txt
new file mode 100644
index 0000000..326d38e
--- /dev/null
+++ b/lab3/qwerty.txt
@@ -0,0 +1,2 @@
+Скриншоты
+
diff --git a/lab3/sort.c b/lab3/sort.c
index 12fe397..addb61a 100644
--- a/lab3/sort.c
+++ b/lab3/sort.c
@@ -1,12 +1,28 @@
-//qwerty
-//qwerty2.0
:

```

13.Объедините ветки

```

segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git merge feature/uppercase
Updating 1b18dbe..5913d27
Fast-forward
 lab3/file1.txt      | 1 +
 lab3/file2.txt      | 1 +
 lab3/greeting.txt   | 2 ++
 lab3/qwerty.txt     | 2 ++
 lab3/sort.c          | 38 ++++++-----
 lab3/sort2.c         | 2 +-
 lab3/text2.txt       | 3 +++
7 files changed, 37 insertions(+), 12 deletions(-)
create mode 100644 lab3/file1.txt
create mode 100644 lab3/file2.txt
create mode 100644 lab3/greeting.txt
create mode 100644 lab3/qwerty.txt
create mode 100644 lab3/text2.txt

```

14. Используйте cat, чтобы увидеть содержимое файла greetings.txt

```

segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ cat greeting.txt
hello
HELLO

```

15. Удалите ветку с заглавными буквами (feature/uppercase)

```

segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git branch -d feature/uppercase
Deleted branch feature/uppercase (was 5913d27).

```

16. Смержьте ветку mybranch в master (git merge)

```

segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git merge mybranch
Already up to date.

```

17. Что выводит git log --oneline --graph --all?

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git log --oneline --graph --all
* 5913d27 (HEAD -> main) HELLO
* cfffc6a (mybranch) Add content to greeting.txt
* 6ac0633 Add file greeting.txt
* efad221 (origin/mybranch) Сделал коммит
* b4e2531 Скрины
* f5f4d8b QWERTY
* 1b18dbe (origin/main, origin/HEAD) qwe
* 5018656 qwerty3.0
* aef3ca6 qwerty2.0
* 4a4da0d Новый файл
* 79375b0 Rename 1_1.c to 1.c
* 30ecf46 Create 2.c
* 7c4cde8 Update README.md
* 3f9d697 Update README.MD
* 0c204fd Update README.MD
* 3dae471 Update README.MD
* 45625a1 Update README.MD
* 207e6de Update README.MD
* 88e4e77 Update README.md
* 08ed5dc Update README.md
* e7169a8 Create README.md
* 80dbcb1 Update README.md
* 64c5395 Update README.md
* feb2432 Update 1_1.c
* 5d6690e Update README.MD
* 0b15131 Delete lab1/ЗАДАНИЕ
* c69011c Create README.MD
* 6f04c60 Rename sort.c to 1_1.c
* e9a11b4 Update sort.c
* 58f0967 Rename ЗАДАНИ to ЗАДАНИЕ
* b58c0f9 Update README.md
* 9f5759c Create README.md
* 0a03bf6 Create ЗАДАНИЕ
* e7daf23 Create sort.c

```

18. Запустить изменения ветки master на удаленный репо.

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git push origin main
Username for 'https://github.com': segoga
Password for 'https://segoga@github.com':
Enumerating objects: 14, done.
Counting objects: 100% (14/14), done.
Delta compression using up to 20 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (12/12), 942 bytes | 39.00 KiB/s, done.
Total 12 (delta 6), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (6/6), completed with 2 local objects.
To https://github.com/segoga/ProgrammingC_2_sibs
1b18dbe..5913d27  main -> main

```


19. Запустить документ с результатами вашей работы

Оценка 5:

Выполнить на оценку 4, а затем попробуем исправить пару merge conflict'ов

- 1) Создать ветку branch1, переключиться на нее

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git branch branch1
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git switch branch1
A      "lab3/\320\237\321\200\320\260\320\272\321\202\320\270\321\207\320\2
65\321\201\320\272\320\276\320\265 \320\267\320\260\320\264\320\260\320\275\
320\270\320\265 3.docx"
Switched to branch 'branch1'
```

- 2) Выполнить команду

```
$ echo "This is a relevant fact" > file.txt
```

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ echo "This is a relevant fact" > file.txt
```

- 3) Закоммитить это изменение

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git add file.txt
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git commit -m "Это известный факт"
[branch1 8346add] Это известный факт
1 file changed, 1 insertion(+)
create mode 100644 lab3/file.txt
```

- 4) Переключиться на главную ветку и выполнить команду

```
echo "This is an indispensable truth!" > file.txt
```

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git switch main
warning: unable to unlink 'lab3/Практическое задание 3.docx': Permission den
ied
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ echo "This is an indispensable truth!" > file.txt
```

- 5) Закоммитить изменения в master

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git add file.txt
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git commit -m "Это что-то там правда"
[main f8f5ecd] Это что-то там правда
1 file changed, 1 insertion(+)
create mode 100644 lab3/file.txt
```

- 6) Каков вывод git log --oneline --graph --all


```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git log --oneline --graph --all
* f8f5ecd (HEAD -> main) Это что-то там правда
| * 8346add (branch1) Это известный факт
| * f2e8714 qwerty
|/
* 5913d27 (origin/main, origin/HEAD) HELLO
* cfffc6a (mybranch) Add content to greeting.txt
* 6ac0633 Add file greeting.txt
* efad221 (origin/mybranch) Сделал коммит
* b4e2531 Скрины
* f5f4d8b QWERTY
* 1b18dbe qwe
* 5018656 qwerty3.0
* aef3ca6 qwerty2.0
* 4a4da0d Новый файл
* 79375b0 Rename 1_1.c to 1.c
* 30ecf46 Create 2.c
* 7c4cde8 Update README.md
* 3f9d697 Update README.MD
* 0c204fd Update README.MD
* 3dae471 Update README.MD
* 45625a1 Update README.MD
* 207e6de Update README.MD
* 88e4e77 Update README.md
* 08ed5dc Update README.md
* e7169a8 Create README.md
* 80dbcb1 Update README.md
* 64c5395 Update README.md
* feb2432 Update 1_1.c
* 5d6690e Update README.MD
* 0b15131 Delete lab1/README

```

- 7) Использовать команду git merge чтобы смержить ветку branch1 в master (получим конфликт это норм)

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git merge branch1
error: The following untracked working tree files would be overwritten by merge:
        lab3/Практическое задание 3.docx
Please move or remove them before you merge.
Aborting

```

- 8) Что показывает git status?

```

segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    "~/320\260\320\272\321\202\320\270\321\207\320\265\321\201\320\272\
320\276\320\265 \320\267\320\260\320\264\320\260\320\275\320\270\320\265 3.d
ocx"
    "\320\237\321\200\320\260\320\272\321\202\320\270\321\207\320\265\32
1\201\320\272\320\276\320\265 \320\267\320\260\320\264\320\260\320\275\320\2
70\320\265 3.docx"

nothing added to commit but untracked files present (use "git add" to track)

```

- 9) Посмотреть содержимое файла file.txt и в любом любимом текстовом редакторе исправить конфликт. Желательно делать это в VSCode или любой другой умной среде, но если нет vscode и не боитесь vim и подобных редакторов, то можно через команду git mergetool.

```

segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git add file.txt
segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git commit -m "wer"
[main 96ff6aa] wer
segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git merge branch1
Already up to date.

```

- 10) Что показывает git log --oneline --graph?

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git log --oneline --graph
* f8f5ecd (HEAD -> main) Это что-то там правда
* 5913d27 (origin/main, origin/HEAD) HELLO
* cfffc6a (mybranch) Add content to greeting.txt
* 6ac0633 Add file greeting.txt
* efad221 (origin/mybranch) Сделал коммит
* b4e2531 Скрины
* f5f4d8b QWERTY
* 1b18dbe qwe
* 5018656 qwerty3.0
* aef3ca6 qwerty2.0
* 4a4da0d Новый файл
* 79375b0 Rename 1_1.c to 1.c
* 30ecf46 Create 2.c
* 7c4cde8 Update README.md
* 3f9d697 Update README.MD
* 0c204fd Update README.MD
* 3dae471 Update README.MD
* 45625a1 Update README.MD
* 207e6de Update README.MD
* 88e4e77 Update README.md
* 08ed5dc Update README.md
* e7169a8 Create README.md
* 80dbcb1 Update README.md
* 64c5395 Update README.md
* feb2432 Update 1_1.c
* 5d6690e Update README.MD

```

11) Запустить изменения

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git push origin main
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 20 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (8/8), 716 bytes | 24.00 KiB/s, done.
Total 8 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 2 local objects.
To https://github.com/seg0ga/ProgrammingC_2_sibsutis
5913d27..d8db151  main -> main

```

Починим Merge конфликты для сортировки MergeSort на python.
Содержимое **base.py**, **lefty.py** и **righty.py** находится в конце этого документа.

1) Находясь в ветке master создадим файл mergesort.py с содержимым из **base.py** (см. приложение)

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ cat mergesort.py
def merge_sort(m):
    """Sort list m, using merge sort"""
    return m.sort() # TODO: Replace with actual implementation
```

- 2) Проиндексируем файл и закоммитим

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git add mergesort.py
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git commit -m "add mergesort.py"
[main 24758ac] add mergesort.py
1 file changed, 3 insertions(+)
create mode 100644 lab3/mergesort.py
```

- 3) Создадим новую ветку Mergesort-Impl и переключимся на нее.

```
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git branch Mergesort-Impl
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git switch Mergesort-Impl
Switched to branch 'Mergesort-Impl'
```

- 4) Содержимое файла mergesort.py заменим на код из righty.py

```

segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ cat mergesort.py
from heapq import merge

def merge_sort2(m):
    """Sort list, using two part merge sort"""
    if len(m) <= 1:
        return m

    # Determine the pivot point
    middle = len(m) // 2

    # Split the list at the pivot
    right = m[middle:]
    left = m[:middle]

    # Sort recursively
    right = merge_sort2(right)
    left = merge_sort2(left)

    # Merge and return
    return list(merge(right, left))

def merge_sort4(m):
    """Sort list, using four part merge sort"""
    if len(m) <= 4:
        return sorted(m)

    # Determine the pivot point
    middle = len(m) // 2
    leftMiddle = middle // 2
    rightMiddle = middle + leftMiddle

```

5) Коммитим изменения.

```

segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git add mergesort.py
segoga@Notebook: /mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git commit -m "new code in mergesort.py"
[Mergesort-Impl 13f173d] new code in mergesort.py
1 file changed, 45 insertions(+), 3 deletions(-)
rewrite lab3/mergesort.py (100%)

```

6) Переключаемся на master и меняем все содержимое mergesort.py на lefty.py

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git switch main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ nano mergesort.py
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ cat mergesort.py
from heapq import merge

def merge_sort2(m):
    """Sort list, using two part merge sort"""
    if len(m) <= 1:
        return m

    # Determine the pivot point
    middle = len(m) // 2

    # Split the list at the pivot
    left = m[:middle]
    right = m[middle:]

    # Sort recursively
    right = merge_sort2(right)
    left = merge_sort2(left)

    # Merge and return
    return list(merge(right, left))

def merge_sort4(m):
    """Sort list, using four part merge sort"""
    if len(m) <= 4:

```

7) Коммитим изменения.

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git add mergesort.py
segoga@Notebook:/mnt/c/Users/Prora/Desktop/Ппора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git commit -m "second new code"
[main 11cddc1] second new code
 1 file changed, 45 insertions(+), 3 deletions(-)
 rewrite lab3/mergesort.py (100%)

```

8) Что показывает git log --oneline --graph?


```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git log --oneline --graph
* 11cddc1 (HEAD -> main) second new code
* 24758ac add mergesort.py
* 96ff6aa (origin/main, origin/HEAD) wer
| \
| * 8346add (branch1) Это известный факт
| * f2e8714 qwerty
* | d8db151 wer
* | f8f5ecd Это что-то там правда
|/
* 5913d27 HELLO
* cfffc6a (mybranch) Add content to greeting.txt
* 6ac0633 Add file greeting.txt
* efad221 (origin/mybranch) Сделал коммит
* b4e2531 Скрины
* f5f4d8b QWERTY
* 1b18dbe qwe
* 5018656 qwerty3.0
* aef3ca6 qwerty2.0
* 4a4da0d Новый файл
* 79375b0 Rename 1_1.c to 1.c
* 30ecf46 Create 2.c
* 7c4cde8 Update README.md
* 3f9d697 Update README.MD
* 0c204fd Update README.MD

```

- 9) Что показывает git branch?

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git branch
  Mergesort-Impl
  branch1
* main
  mybranch

```

- 10) Необходимо сменить Mergesort-Impl в master.

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git merge Mergesort-Impl
Already up to date.

```

- 11) После исправления всех merge конфликтов запустить в master изменения.

```

segoga@Notebook:/mnt/c/Users/Prora/Desktop/Прора/lab_git/ProgrammingC_2_sibs
utis/lab3$ git push origin main
Enumerating objects: 18, done.
Counting objects: 100% (18/18), done.
Delta compression using up to 20 threads
Compressing objects: 100% (16/16), done.
Writing objects: 100% (16/16), 1.76 KiB | 94.00 KiB/s, done.
Total 16 (delta 11), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (11/11), completed with 2 local objects.
To https://github.com/segoga/ProgrammingC_2_sibs
96ff6aa..1ae3ebe main -> main

```