Report

Part C

The goal of part c is to control robot by [UP],[DOWN],[LEFT],[RIGHT] keyboard.Hence,we need to set keyhandler to detect whether keyboard event is activated.

```
ArKeyHandler keyHandler;
Aria::setKeyHandler(&keyHandler);
robot.attachKeyHandler(&keyHandler);
```

Then,I attach keyhandler corresponding event and these event is mainly to adjust rotate velocity and velocity.

```
keyHandler.addKeyHandler(256, new ArGlobalFunctor2<ArRobot*, int*>(&ac
keyHandler.addKeyHandler(257, new ArGlobalFunctor2<ArRobot*, int*>(&de
keyHandler.addKeyHandler(258, new ArGlobalFunctor2<ArRobot*, int*>(&tu
keyHandler.addKeyHandler(259, new ArGlobalFunctor2<ArRobot*, int*>(&tu
```

And because we need to let rotate velocity get zero as key release so I devise a while loop to let rotate velocity get zero in a fixed time.

```
while(true){
  robot.setRotVel(0);
  ArUtil::sleep(1500);
}
```

Part D

In this part we need to detect obstacle and avoid it.So,I use currentReadingPolar function to detect obstacle range and minus it with robot's radius.If this distance shorter than a threshold we set velocity to zero.Otherwise,we set all velocity to multiply 0.2 to constrain fast speed which is hard to stop.

```
while(true){
  robot.setVel(speed*0.2);
  double range = sonarDev.currentReadingPolar(-70, 70)-radius;
  if (range < myStopDistance)
{
  robot.setVel(0);
}

  robot.setRotVel(0);
  ArUtil::sleep(800);
}
```

Part E

(1) They are different

In true pose,the robot location(x,y) is defined by the map

the robot orientation is defined by the map(east is 0,west is 180/-180)

In odometry,the robot location(x,y) is define by the robot initial run time location,and the initial point is (0,0)

the robot orientation is defined by the robot initial run time orientation,and the initial orientation is 0 turn left increase degree and turn right decrease degree

(2)First we need to set robot to a true position so we send packet to know true position.

```
ArRobotPacket pkt;
pkt.setID(ArCommands::SIM_SET_POSE);
pkt.uByteToBuf(0);
pkt.byte4ToBuf(5090);
pkt.byte4ToBuf(3580);
pkt.byte4ToBuf(3093.97);
pkt.finalizePacket();
robot.getDeviceConnection()->write(pkt.getBuf(),pkt.getLength());
```

Then,we use move() and setHeading() to move to target position

```
robot.lock();
robot.setHeading(0);
robot.unlock();

while(1)
{
  robot.lock();
  if(robot.isHeadingDone())
  {
      robot.unlock();
      break;
   }
  robot.unlock();
    ArUtil::sleep(100);
}

  robot.lock();
  robot.move(x-5090);
  robot.unlock();

while(1)
{
  robot.lock();
   if(robot.isMoveDone())
   {
      robot.unlock();
      break;
   }
    robot.unlock();
    ArUtil::sleep(50);
}
```

```
  robot.lock();
  robot.setHeading(90);
  robot.unlock();


while(1)
{
  robot.lock();
   if(robot.isHeadingDone())
   {
      robot.unlock();
      break;
   }
  robot.unlock();
  ArUtil::sleep(100);
}

  robot.lock();
  robot.move(y-3580);
  robot.unlock();

while(1)
{
  robot.lock();
   if(robot.isMoveDone())
   {
      robot.unlock();
      break;
   }
    robot.unlock();
    ArUtil::sleep(50);
}
```

Last,turn the robot to the right orientation

```
robot.lock();
robot.setHeading(theta);
robot.unlock();
```

Demo link:

Partc :

https://drive.google.com/file/d/15c670vwDkEzeBKPWDBfg2v_rBNE5fZAj/view?usp=sharing

Partd :

https://drive.google.com/file/d/1DHywQ6RCh1fQuH_qCcweysaqyF2l9wFK/view?usp=sharing

Parte :

https://drive.google.com/file/d/1m9bgGJML4qQfnybrOzByU-YWF2Feacqk/view?usp=sharing