



# پیاده سازی پروژه شبیه سازی شبکه (دانشگاه صنعتی ارومیه)

نام تیم : AMI

اعضای تیم :

امین اتحادی

محمد شریفی

نام دانشگاه : دانشگاه صنعتی ارومیه

ترم دوم

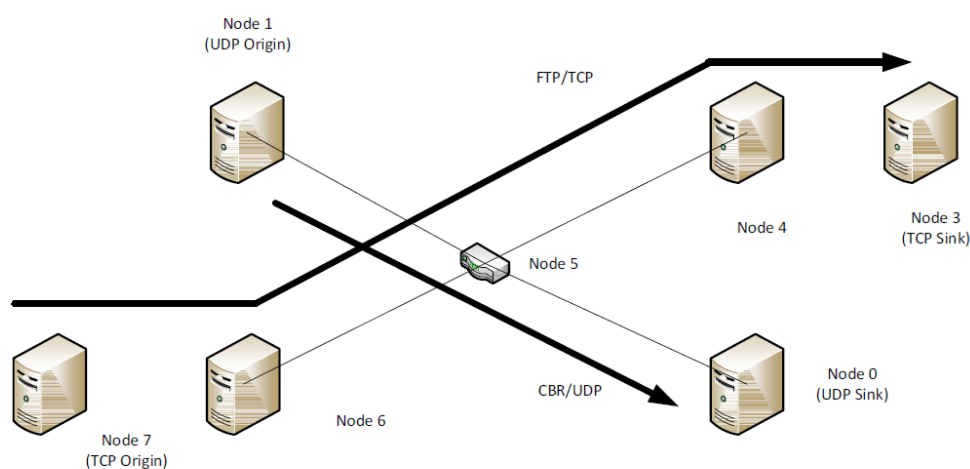
۹۳-۹۴

## فهرست مطالب

۴	مقدمه
۵	معرفی گروه
۶	پیاده سازی پروژه در ns-2
۸	نمایش نحوه اجرا
۱۱	پروژه برنامه نویسی
۱۴	نتیجه گیری
۱۵	منابع

## پروژه شبیه سازی

- شبکه ای متشکل از 6 گره به شرح زیر را در نظر بگیرید. ساختار شبکه را در یک شبیه سازی با نرم افزار NS2 ایجاد کنید. تمام لینک ها با ظرفیت 10 مگابیت بر ثانیه هستند.



برای شبیه سازی شبکه بالا از نرم افزار ns در سیستم عامل لینوکس استفاده می نماییم. لازم به ذکر است که در شبکه بالا شماره گذاری نودها اشتباه است. چنان که در کد برنامه شماره گذاری شکل بالا در نظر گرفته شده است.

## معرفی گروه

مهندس امین اتحادی و مهندس محمد شریفی دانشجویان مهندسی کامپیوتر دانشگاه صنعتی ارومیه پیاده سازی و اجرای این پروژه را بر عهده داشته اند. امید است رضایت استفاده کنندگان از این پروژه را جلب نمایید.

## با تشکر

محمد شریفی

امین اتحادی

در سیستم عامل لینوکس با استفاده از یکی از **Editor** موجود مراحل زیر را مرحله به مرحله پیاده سازی می کنیم.(برای آشنایی و یادگیری بیشتر می توانید به **pdf** شبیه سازی شبکه های کامپیوتری " آشنایی با بسته نرم افزاری ns-2 " نویسنده: آقای سیامک عضدالملکی مراجعه کنید). سپس با وارد کردن دستور: **tcl**. نام پروژه **ns** ، در ترمینال دو فایل با پسوندهای **nam** و **tr**. ایجاد میشود. که در ادامه با کار **tr**. کار خواهیم نمود و برای تحلیل ترافیک شبکه این فایل را به وسیله برنامه ای که در زبان **C++** پیاده سازی شده است ، برخی از اطلاعات مورد نیاز را استخراج نموده و به صورت فایلی با پسوند **CSV**. فرمت اکسل در آورده و ذخیره می نماییم.

مراحل زیر را به ترتیب پیاده سازی می کنیم :

- 1.#Create a simulator object
- 2.#Define different colors for data flows (for NAM)
- 3.#Open the NAM trace file
- 4.#Open the Trace file
- 5.#Define a 'finish' procedure
  - 5-1. #Close the NAM trace file
  - 5-2.#Execute NAM on the trace file
- 6.#Create sevsn nodes
7. #Create links between the nodes
8. #Set Queue Size of link (n1-n5) to 10
9. #Give node position (for NAM)
10. #Monitor the queue for link (n1-n5). (for NAM)

10-1. # \$ns duplex-link-op \$n1 \$n5 queuePos 0.5

11. #Setup a TCP connection

12. #Setup a FTP over TCP connection

13. #Setup a UDP connection

14. #Setup a CBR over UDP connection

15. #Schedule events for the CBR and FTP agents

16. #Detach tcp and sink agents (not really necessary)

17. #Call the finish procedure after 5 seconds of simulation time

18. #Print CBR packet size and interval

19. #Run the simulation

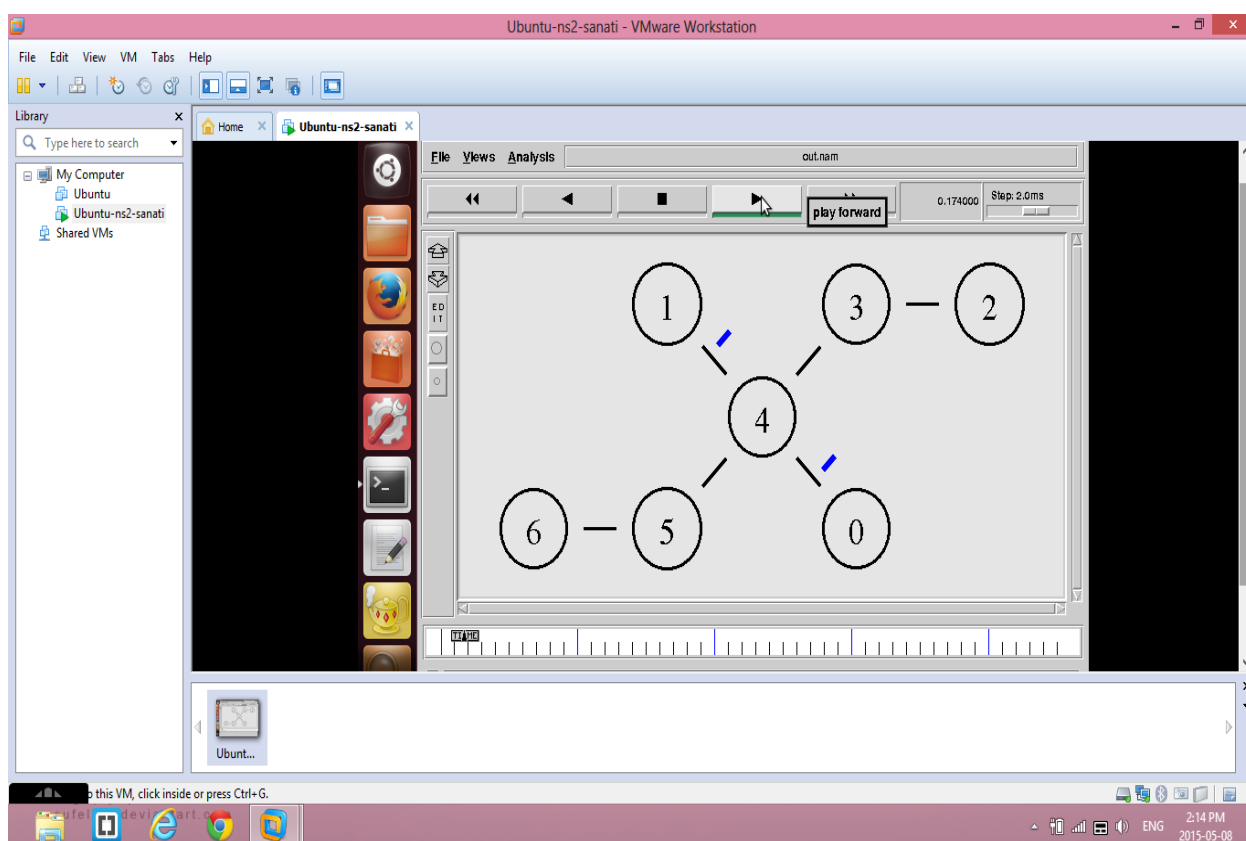
برای شروع پیاده سازی ابتدا رنگ پکت های رد و بدل شده را در ابتدای برنامه مشخص می کنیم. (سبز، آبی)، در ادامه تابع مربوط به اجرای برنامه که منجر به تولید فایل های tr و nam. می شود را می نویسیم، سپس به تعداد ایستگاه های مورد نیاز در برنامه نود تعریف می کنیم، سپس لینک های ارتباطی مابین نودها و ظرفیت عبوری از هر لینک ارتباطی را تعریف می کنیم و پس از آن محل نودهای را در برنامه مشخص می کنیم. و در ادامه نوع پروتکل های استفاده شده در بین ایستگاه ها را تعریف می نماییم که در برنامه زیر کاملاً مشخص است. سپس زمان شروع و پایان ارسال را برای هر یک از پروتکلها و مشخص کرده و اندازه پکت ها را نیز مشخص می نماییم.

کد های مربوط به هر مرحله در فایل networkproject.tcl به صورت واضح موجود است.

## نمایش نحوه اجرا

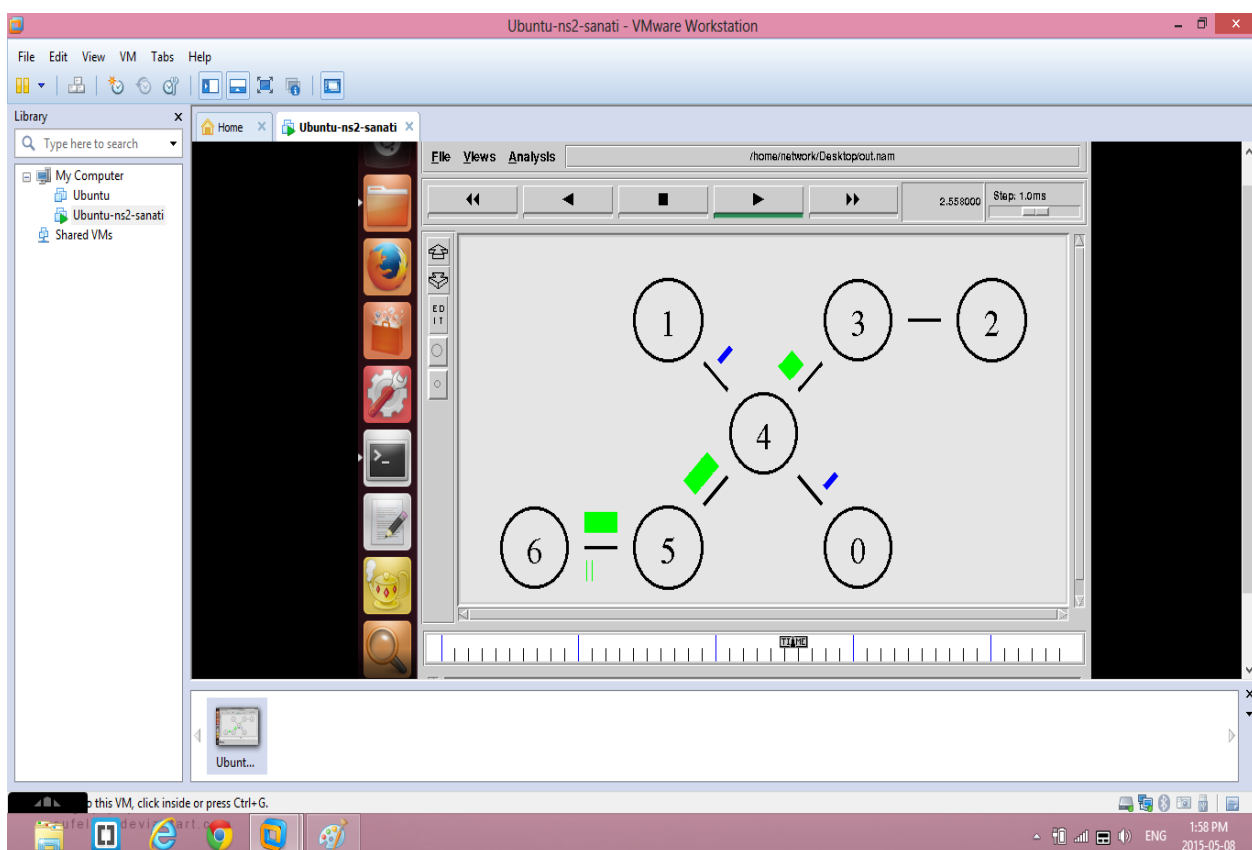
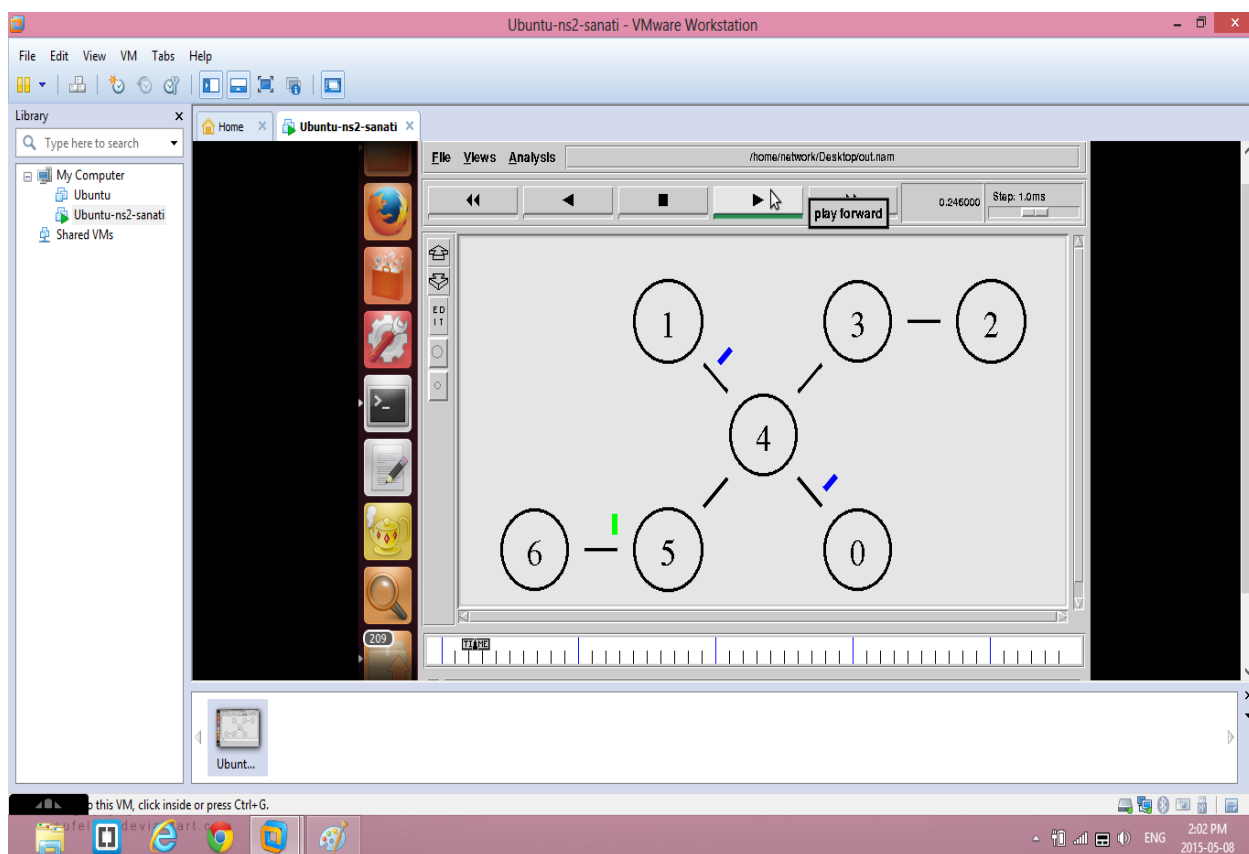
پس از اجرای برنامه شکل شبکه پیاده سازی شده به صورت زیر به نمایش در خواهد آمد.

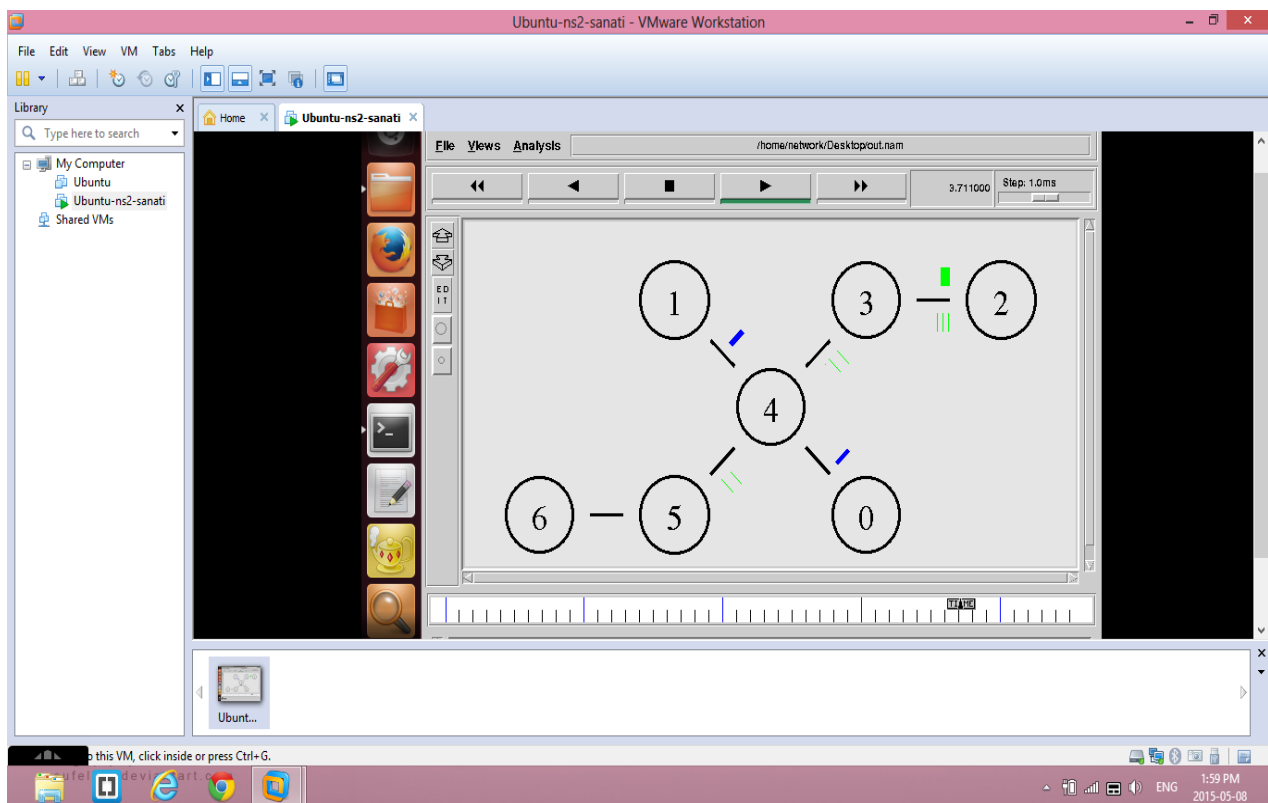
با کلیک بر روی دکمه **play forward** ابتدا ارسال اطلاعات از نود ۱ به سمت نود ۰ با پروتکل **cbr/udp** آغاز می شود.



پس از گذشت زمان ارسال اطلاعات از نود ۶ به سمت نود ۲ با پروتکل **ftp/tcp** نیز آغاز می شود.

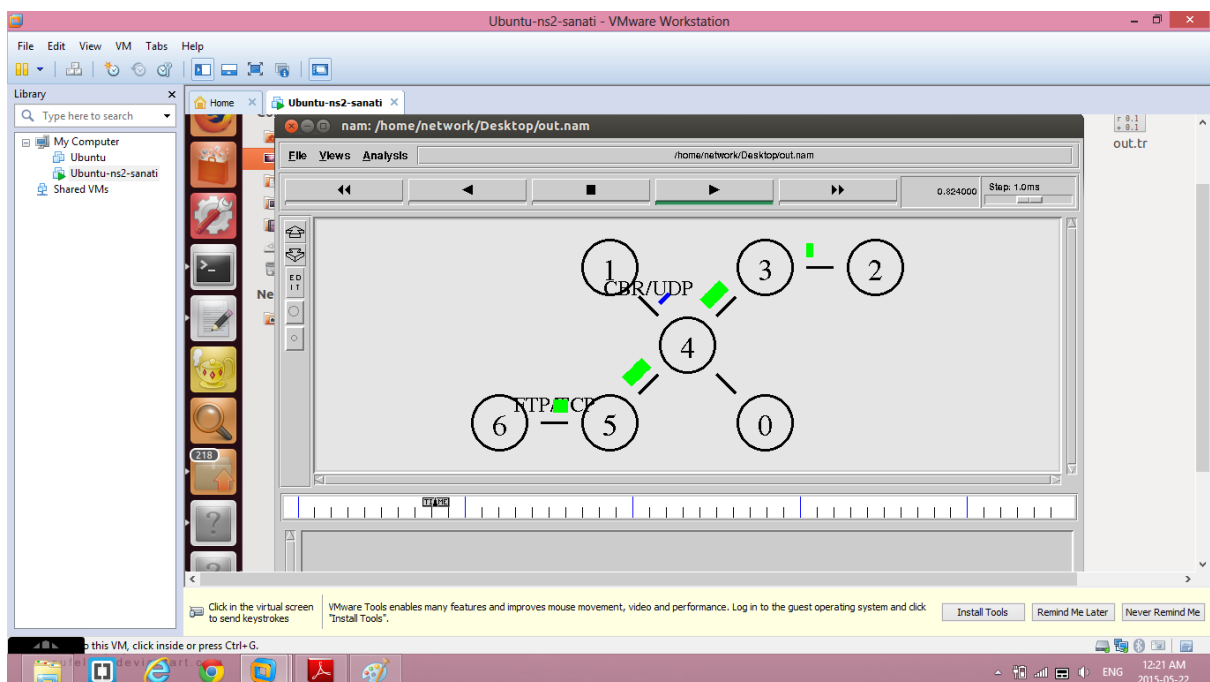






این روند تا پایان زمان اجرای برنامه ادامه می یابد. البته این نرم افزار دارای امکانات دیگری نیز می باشد. که در صورت لزوم میتوان از آنها استفاده نمود.

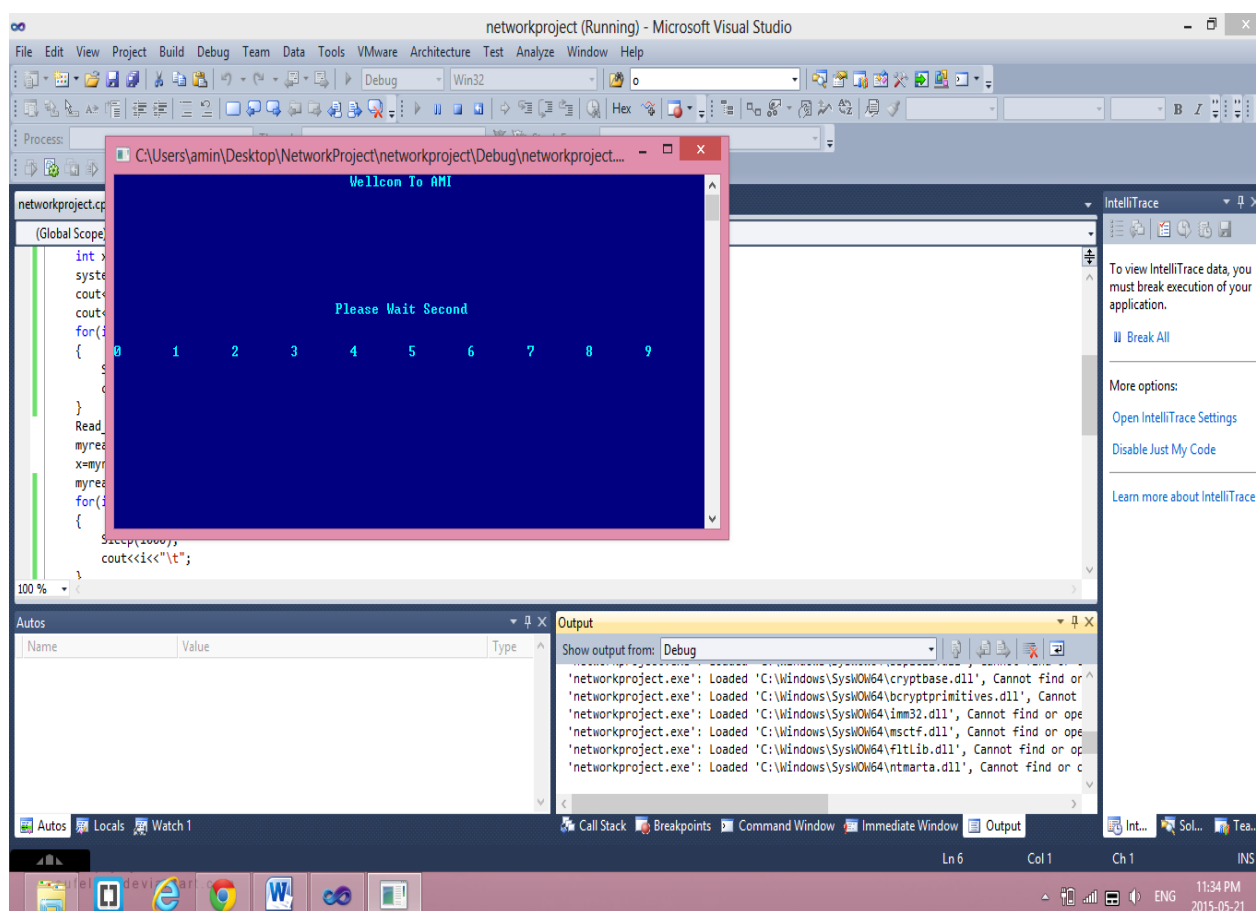
لازم به ذکر است نام پروتکل ها را نیز به پروژه اضافه نموده ایم.



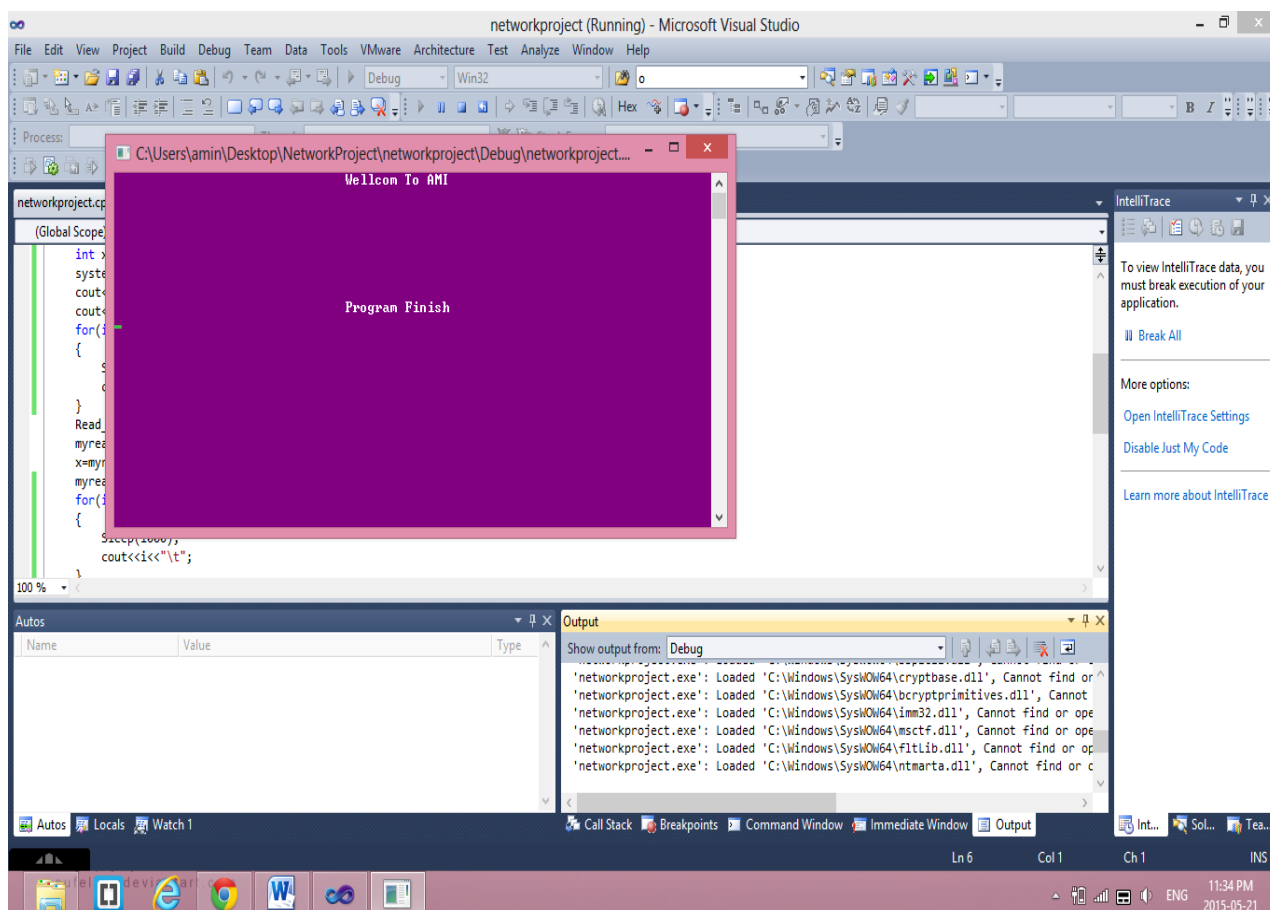
برنامه به نام **networkproject** در پوشه پروژه موجود می باشد.

برنامه خواسته شده به وسیله زبان **C++** پیاده سازی شده است. و از توابع کتابخانه ای برای ایجاد فایل، خواندن از فایل و نوشتن در فایل در آن استفاده شده است. که در این برنامه در کلاس **Read\_File** به وسیله تابعی که در آن تعریف شده است تعداد خطوط فایل به دست می آید و به تابع بعدی فرستاده می شود. و سپس تمام محتویات فایل **out.tr** در یک آرایه دو بعدی ذخیره می شود. و پس از استخراج اطلاعات مورد نیاز از فایل مبدا (اطلاعاتی را از فایل مبدا در فایل مقصد می ریزد که در قسمت ذخیره فایل در داخل حلقه **for** به صورت شرط **if** آمده است)، آن اطلاعات در فایل مقصد که در ادامه نحوه دسترسی و محل این فایل را (که در پوشه برنامه می باشد) توضیح خواهیم داد، قابل دسترسی می باشد. کد برنامه بسیار واضح می باشد که در اینجا از توضیح جزئیات آن خودداری می کنیم.

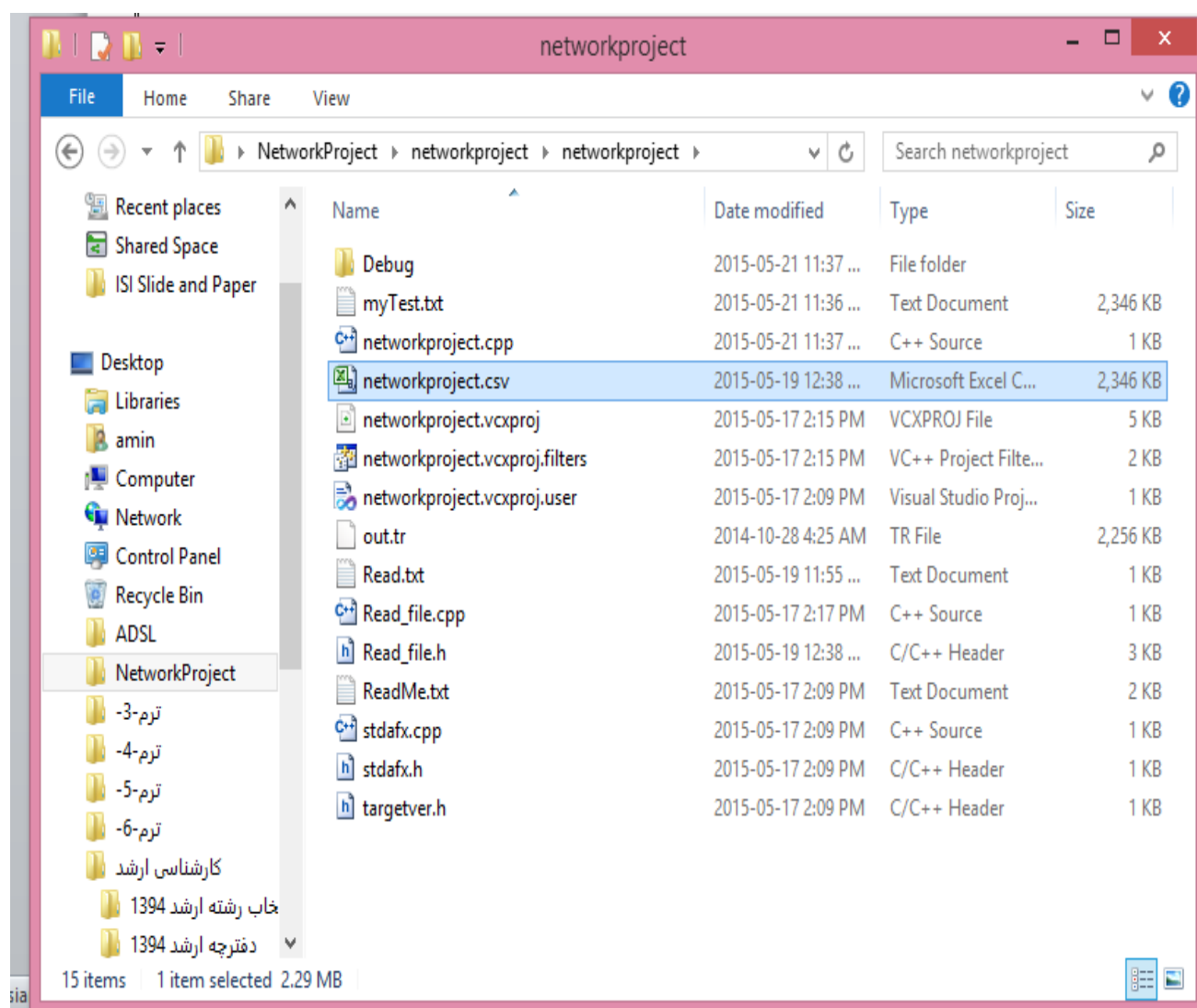
پس از اجرای برنامه شکل زیر به نمایش در خواهد آمد:



که پس از چند ثانیه شکل زیر پایان برنامه را اعلام می کند با فشردن کلید Enter از برنامه خارج می شوید:



فایل خروجی که در اکسل قابل نمایش است (با پسوند CSV). در داخل پوشه با نام networkproject ذخیره شده است در شکل زیر محل فایل نشان داده شده است. (برای راحتی بیشتر فایل خروجی در پوشه اصلی پروژه ارسال شده نیز موجود است).



پایان...

## نتیجه گیری

.....

